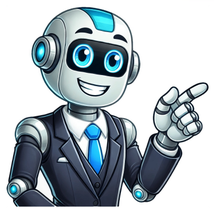
Continue



Skip to content Accessibility Policy Oracle Java is the #1 programming language and development platform. It reduces costs, shortens development timeframes, drives innovation, and improves application services. With millions of developers running more than 60 billion Java Virtual Machines worldwide, Java continues to be the development platform of choice for enterprises and developers. Assess the health of your Java environment Java 24 is now available The next Java release improves the performance, stability, and security of Java application development. Oracle GraalVM free on OCI Build native executables that help applications start up fast, reduce memory usage, and save hosting costs. Java Management Service JMS Advance features help administrators gain additional insights into Java workloads. Analyze usage, vulnerabilities, and impact from Cryptographic Roadmap updates. Transforming Development for Next-Generation Software Supply Chains Get executive insights on leading trends and challenges impacting development organizations today, along with technology choices such as Java, that can help address them efficiently based on a recent VDC Research study. Get the VDC Research report Top IT security and compliance pain points in application development Security remains the most important priority for IT executives, according to a recent report from 451 Research. Read the brief to find out the top security concerns for developers and how you can achieve your IT security and compliance goals with Java. Get the 451 Research brief Aberdeen Knowledge Brief Learn why leaders in application development use Java to more efficiently build world-class applications the highest quality and the strongest security. Case study: Oracle Java EPP for Oracle Fusion Find out why Oracle Java Enterprise Performance Pack helps Oracle Fusion Applications improve application response times by 40% and decrease CPU utilization by 25%. Read the case study (PDF) Take advantage of the high-performance JDK with advanced optimizations that improve Java application performance and microservices deployment—on-premises and in the cloud. Together, the two technologies add value for cloud native deployments with native image and multilanguage support. Features Advanced optimizing just-in-time compiler Ahead-of-time compiler (compile Java native executables) Seamless interoperability for polyglot applications Built on enterprise-class Oracle Java SE 24/7 Oracle Premiere Support (My Oracle Support) Oracle Cloud Infrastructure (OCI) enhances the versatility, power, and stability of Java. As the steward and leading contributor to the Java platform, Oracle continues to drive the evolution of Java in response to the demands of enterprises and to provide unparalleled expertise to support developers. Explore Java on Oracle Cloud Infrastructure Enhanced Java performance. Boost peak performance with high-performance JDK. Build and deploy native executables that start almost instantly and use fewer memory resources. Simplified Java Management Service gives you insights into all your Java deployments, on OCI and on-premises. Expert Java support from the stewards of Java to optimize Java apps whenever it makes sense. Exclusive access to patches and updates, even beyond the end of public updates. Benefits of Java SE on OCI at no additional cost. Java Card enables secure elements, such as smart cards and other tamper-resistant security chips, to host applications based on Java technology. Store and update multiple applications on a single, resource-constrained device. Features Interoperable Secure execution environment Multiapplication, multitenant Extensible and updatable “One of our largest customers was facing performance and memory issues after upgrading their system. There was huge pressure from the customer at all levels to resolve the issue as soon as possible. Once the Java Sustaining Engineering team got involved, collaborating with Oracle Support, they were able to pinpoint what was causing the issue and what changes to make in order to resolve the issue. This solution was provided very quickly, and we received kudos from the end user’s executives as to the speed and efficacy of the solution. We greatly appreciate the efforts of the Java Sustaining Engineering Team and Oracle Customer Support.” Takashi Hashizume, Senior Manager, AI Platform Division, NEC Corporation MIKS Limited Computer Software Company "Oracle Java SE Subscription's multilingual support team is very experienced and readily available to provide instant and parallel support helping our developers to build their projects more timely and easily with no hassles." —Mohammad Iqbal Khan, Project Manager Rothbadi & Co. IT Services "Instead of wasting time and money, we have been able to reduce overall costs by managing our Java estate with Oracle Java SE Subscription. This gives us a huge cost saving opportunity that significantly reduces our IT OPEX bills." —Fortune Nwaiwu, Business Analyst "For our professional customers, where reliability is a top priority, the small fee of Oracle Jave SE subscription is much more valuable than the many free platforms available for the returned value. We use it for many deployments, mostly for high-value solutions, where every small detail matters." —Balázs Kiss, Software Developer Corte Suprema de Justicia "The technical support Oracle provides is highly efficient and of very good quality. Their staff is trained and has the necessary experience to solve or guide in the resolution of problems raised." —Moris Mendez, Ing. de Sistemas Informaticos March 18, 2025 Sharat Chander | Senior Director, Java Product Management and Developer Relations Learn about the Java 24 release and ongoing Java innovations that address modern application development. Read the complete post | Subscribe to the blog Manage Java SE installations, updates, and upgrades across your enterprise more cost effectively. Discover the advantages of a Java license and support from the Java experts who wrote the code. Talk to a Java team member about the advantages of an Oracle Java SE Subscription. Java is one of the best object-oriented programming language developed by James Gosling from Sun Microsystems in the year 1991 and was publicly available in the year 1995. It is an interpreted programming language with platform independency making it one of the best programming language among all. In this article, we'll see some interesting and cool tricks in Java. Executing Comments: Most of the developers think comments are never executed in a program and are used for ease in understanding the code. But, they are executed. For example: Java public class GFG { public static void main(String[] args) { // \u000d System.out.println("GeeksForGeeks"); } } Explanation: This comment is executed because of Unicode character "\u000d" and java compiler parses this unicode character as a new line. Java allows using Unicode characters without encoding.Underscore in Numeric Literals: In Java SE 7 and above, underscores can be used in numeric literals without generating any warning or error in the output. Example: Java public class GFG { public static void main(String[] args) { int x = 123_34; System.out.println(x); } } Double Brace Initialization: In Java, collections such as sets, lists, maps, etc. does not have a simple and easy way to initialize the values during declaration. Developers either push values into the collection or creates a static block for the constant collection. Using double brace initialization, collections can be initialized during declaration with less efforts and time. Example: Java import java.util.HashSet; import java.util.Set; public class GFG { public static void main(String[] args) { Set GFG = new HashSet() {{ add("DS"); add("ALGORITHMS"); add("BLOCKCHAIN"); add("MACHINE LEARNING"); } }; System.out.println(GFG); } } Output[MACHINE LEARNING, ALGORITHMS, DS, BLOCKCHAIN]Finding a position to insert the numeric element in the array: There is a small cool trick to find the position where the requested element can be inserted in the sorted array. Example: Java import java.util.Arrays; public class GFG { public static void main(String[] args) { int[] arr = new int[] { 1, 3, 4, 5, 6 }; // 2 has to be inserted int pos = Arrays.binarySearch(arr, 2); System.out.print("Element has to be inserted at: " + ~pos); } } OutputElement has to be inserted at: 1Wrapper class vs datatype: In the below example, the second print statement will not display true because reference of wrapper class objects are getting compared and not their values. Java import java.util.Arrays; public class GFG { public static void main(String[] args) { int num_1 = 10; int num_2 = 10; Integer wrapnum_1 = new Integer(10); Integer wrapnum_2 = new Integer(10); System.out.println(num_1 == num_2); // Compares reference System.out.println(wrapnum_1 == wrapnum_2); // Compares value of object System.out.println(wrapnum_1.equals(wrapnum_2)); } } Disclosure: Hackr.io is supported by its audience. When you purchase through links on our site, we may earn an affiliate commission. Embarking on a coding journey in the realm of Java? Exciting times await as you delve into the world of Java projects! This programming powerhouse provides a diverse array of Java projects to sharpen your skills. Unsure where to begin? Fear not! We’ve meticulously curated a collection of engaging and imaginative Java projects, ensuring a thrilling and educational coding experience. Explore the possibilities and kickstart your Java Projects adventure today!Java isn’t just a programming language; it’s the force behind mobile apps, desktop software, web servers, gaming experiences, and seamless database connections. So, if your goal is to step into the realm of development, it’s time to dive into coding.Building a portfolio with real-world projects is the key to showcasing your skills. Ready to roll up your sleeves? We’ve handpicked the top 10 Java projects for beginners and not only in 2024 to kickstart your coding adventure!Java is a versatile and widely-used programming language known for its portability, flexibility, and security features. Developed by Sun Microsystems (now owned by Oracle Corporation), Java was released in 1995 and has since become one of the most popular programming languages in the world.Key characteristics of Java include:CharacteristicDescriptionPlatform IndependenceWrite once, run anywhere. Code written in Java can run on any device or platform with a Java Virtual Machine (JVM).Object-OrientedEmphasizes the use of classes and objects, promoting modular and reusable code.Robust and SecureFeatures automatic memory management (garbage collection), strong type-checking, and security measures like sandboxing.Multi-threading SupportAllows developers to write programs that can perform multiple tasks concurrently.Rich Standard LibraryComprehensive library providing pre-built functionality for various common tasks.Community and EcosystemLarge and active developer community contributing to a diverse ecosystem of libraries, frameworks, and tools.VersatilityUsed in web development, mobile app development (Android), enterprise applications, cloud computing, and more.Java’s characteristics make it a versatile and powerful language for a wide range of applications and development scenarios.Dive into graphical user interface (GUI) development with this simple project. Display a friendly “Hello, World!” message using Java Swing, laying the foundation for more interactive applications. import javax.swing.*; public class HelloWorldGUI { public static void main(String[] args) { // Create a JFrame (a window) JFrame frame = new JFrame("Hello World GUI"); // Create a JLabel (a text label) JLabel label = new JLabel("Hello, World!"); // Set the default close operation for the JFrame frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); // Add the JLabel to the JFrame frame.getContentPane().add(label); // Set the size of the JFrame frame.setSize(300, 200); // Make the JFrame visible frame.setVisible(true); } } import javax.swing.*; This line imports the necessary classes from the Swing library, which is used for creating graphical user interfaces in Java. JFrame frame = new JFrame("Hello World GUI"); A JFrame is created, representing the main window of the GUI application. The constructor parameter is the title of the window. JLabel label = new JLabel("Hello, World!"); A JLabel is created, representing a text label. The constructor parameter is the text to be displayed.Set Default Close Operation: frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); This line sets the default close operation for the JFrame. EXIT_ON_CLOSE ensures that the application exits when the window is closed. frame.getContentPane().add(label); The getContentPane() method retrieves the content pane of the JFrame, and add(label) adds the JLabel to it. frame.setSize(300, 200); This line sets the size of the JFrame to be 300 pixels wide and 200 pixels tall. frame.setVisible(true); Finally, this line makes the JFrame visible on the screen.When you run this program, a window titled “Hello World GUI” will appear, displaying the text “Hello, World!” within it. This project serves as a foundational introduction to GUI development in Java, where you can explore more advanced features and build more interactive applications.Build a console-based number guessing game where the computer generates a random number, and the player tries to guess it.Code Snippet: import java.util.Scanner; import java.util.Random; public class NumberGuessingGame { public static void main(String[] args) { // Initialize Scanner for user input and Random for generating a random number Scanner scanner = new Scanner(System.in); Random random = new Random(); // Generate a random number between 1 and 100 (inclusive) int targetNumber = random.nextInt(100) + 1; // Initialize variable to store user's guess int guess; // Start a do-while loop to allow multiple guesses until the correct number is guessed do { // Prompt the user to enter their guess System.out.print("Enter your guess: "); guess = scanner.nextInt(); // Check if the guess is too low, too high, or correct if (guess < targetNumber) { System.out.println("Too low! Try again."); } else if (guess > targetNumber) { System.out.println("Too high! Try again."); } else { System.out.println("Congratulations! You guessed the correct number."); } } while (guess != targetNumber); // Continue the loop until the correct number is guessed } } Explanation:Imports:The import statements bring in the necessary classes (Scanner and Random) from the java.util package.Initialization:Scanner scanner = new Scanner(System.in);: Creates a Scanner object to read user input from the console.Random random = new Random();: Creates a Random object for generating a random number.Generate Target Number:int targetNumber = random.nextInt(100) + 1;: Generates a random number between 1 and 100 (inclusive) as the target number for the user to guess.User Input and Game Loop:int guess;: Declares a variable to store the user’s guess.The do-while loop continues until the user correctly guesses the target number.Inside the loop:System.out.print("Enter your guess: ");: Prompts the user to enter their guess.guess = scanner.nextInt();: Reads the user’s input as an integer.The program provides feedback based on the user’s guess being too low, too high, or correct.Loop Exit:} while (guess != targetNumber);: The loop continues until the user’s guess matches the randomly generated target number.The code efficiently handles user input, generates a random target number, and guides the user through the Number Guessing Game until they successfully guess the correct number.Develop a basic calculator that can perform addition, subtraction, multiplication, and division.Code Snippet: import java.util.Scanner; public class SimpleCalculator { public static void main(String[] args) { Scanner scanner = new Scanner(System.in); System.out.print("Enter first number: "); double num1 = scanner.nextDouble(); System.out.print("Enter second number: "); double num2 = scanner.nextDouble(); System.out.print("Enter operator (+, -, *, /): "); char operator = scanner.next().charAt(0); double result = 0; switch (operator) { case '+': result = num1 + num2; break; case '-': result = num1 - num2; break; case '*': result = num1 * num2; break; case '/': result = num1 / num2; break; default: System.out.println("Invalid operator"); } System.out.println("Result: " + result); } } Explanation:Import Statement:import java.util.Scanner;: Imports the Scanner class from the java.util package for user input.Initialization:Scanner scanner = new Scanner(System.in);: Creates a Scanner object to read user input from the console.User Input:Prompts the user to enter the first and second numbers and the operator:System.out.print("Enter first number: ");double num1 = scanner.nextDouble();System.out.print("Enter second number: ");double num2 = scanner.nextDouble();System.out.print("Enter operator (+, -, *, /): ");char operator = scanner.next().charAt(0);Calculation:Uses a switch statement to perform the calculation based on the entered operator:switch (operator) { ... }Switch Cases:For each case (+, -, *, /), the corresponding operation is performed and the result is stored in the result variable.Default Case:The default case handles an invalid operator and informs the user.Display Result:System.out.println("Result: " + result);: Prints the result of the calculation to the console.This code provides a simple console-based calculator that takes two numbers and an operator from the user, performs the requested operation, and displays the result. The use of a switch statement makes the code concise and easy to understand for basic arithmetic operations.Description: Build a simple to-do list application that allows users to add, edit, and delete tasks.Code Snippet: import java.util.ArrayList; import java.util.Scanner; public class ToDoList { public static void main(String[] args) { ArrayList tasks = new ArrayList(); Scanner scanner = new Scanner(System.in); while (true) { System.out.println("1. Add Task"); System.out.println("2. Edit Task"); System.out.println("3. Delete Task"); System.out.println("4. View Tasks"); System.out.println("5. Exit"); System.out.print("Enter your choice: "); int choice = scanner.nextInt(); switch (choice) { case 1: System.out.print("Enter task: "); String task = scanner.next(); tasks.add(task); break; case 2: // Add logic for editing a task break; case 3: // Add logic for deleting a task break; case 4: System.out.println("Tasks: " + tasks); break; case 5: System.exit(0); default: System.out.println("Invalid choice"); } } } } Explanation:Import Statement:import java.util.ArrayList;: Imports the ArrayList class from the java.util package for dynamic list storage.Initialization:ArrayList tasks = new ArrayList();: Creates an ArrayList to store tasks.Scanner scanner = new Scanner(System.in);: Creates a Scanner object for user input.Menu Loop:The program enters an infinite loop (while (true)) to repeatedly display the task management menu and process user input.Menu Options:Options 1-5 are presented to the user for adding tasks, editing tasks (not implemented), deleting tasks (not implemented), viewing tasks, and exiting the program.User Input:System.out.print("Enter your choice: ");: Prompts the user to enter their choice.int choice = scanner.nextInt();: Reads the user’s choice as an integer.Switch Statement:Uses a switch statement to perform actions based on the user’s choice.Menu Options Explained:Option 1: Adds a new task to the list.Option 2: Placeholder for logic to edit a task (not implemented in the provided code).Option 3: Placeholder for logic to delete a task (not implemented in the provided code).Option 4: Displays all tasks in the list.Option 5: Exits the program.This code provides a simple console-based ToDoList application where users can add tasks, view tasks, and perform other basic operations. The code structure and use of the ArrayList demonstrate basic task management functionality.Create a program that can encrypt and decrypt text files using a simple algorithm (e.g., Caesar cipher).Code Snippet: import java.io.*; public class FileEncryption { public static void main(String[] args) { // Implement file encryption/decryption logic // using FileInputStream, FileOutputStream, and a simple algorithm // Step 1: Define the file paths for input and output String inputFile = "input.txt"; String encryptedFile = "encrypted.txt"; String decryptedFile = "decrypted.txt"; try { // Step 2: Open FileInputStream for reading from the input file FileInputStream inputFileStream = new FileInputStream(inputFile); // Step 3: Open FileOutputStream for writing to the encrypted file FileOutputStream encryptedFileStream = new FileOutputStream(encryptedFile); // Step 4: Implement your encryption algorithm // (Read bytes from inputFileStream, apply encryption, and write to encryptedFileStream) // Step 5: Close FileInputStream and FileOutputStream inputFileStream.close(); encryptedFileStream.close(); // Step 6: Open FileInputStream for reading from the encrypted file FileInputStream encryptedFileStream = new FileInputStream(encryptedFile); // Step 7: Open FileOutputStream for writing to the decrypted file FileOutputStream decryptedFileStream = new FileOutputStream(decryptedFile); // Step 8: Implement your decryption algorithm // (Read bytes from encryptedFileStream, apply decryption, and write to decryptedFileStream) // Step 9: Close FileInputStream and FileOutputStream encryptedFileStream.close(); decryptedFileStream.close(); } catch (IOException e) { e.printStackTrace(); } } } The provided Java code for FileEncryption is a template for a program designed to perform file encryption and decryption using FileInputStream and FileOutputStream. While the actual encryption/decryption logic is not implemented in the code snippet, it provides a structure for you to implement a simple algorithm for these operations.Here’s an explanation of the code structure and the general steps you would need to follow:File Paths:String inputFile = "input.txt";String encryptedFile = "encrypted.txt";String decryptedFile = "decrypted.txt";Define the paths for the input file, encrypted file, and decrypted file.FileInputStream and FileOutputStream:FileInputStream inputFileStream = new FileInputStream(inputFile);FileOutputStream encryptedFileStream = new FileOutputStream(encryptedFile);FileInputStream encryptedFileStream = new FileInputStream(encryptedFile);FileOutputStream decryptedFileStream = new FileOutputStream(decryptedFile);Open streams for reading/writing from/to files.Encryption/Decryption Algorithm:The actual encryption and decryption logic should be implemented between opening and closing the streams.Read bytes from the input file stream, apply encryption/decryption, and write to the output file stream.Exception Handling:Wrap file I/O operations in a try-catch block to handle potential IOExceptions.This code template provides a basic structure for file encryption and decryption. You need to implement the specific algorithm for encrypting and decrypting data according to your requirements. Keep in mind that a secure encryption algorithm should be used in a real-world scenario.Server Side (ChatServer.java): import java.io.*; import java.net.*; import java.util.*; public class ChatServer { private static final int PORT = 5555; private static final List clients = new ArrayList(); public static void main(String[] args) { try (ServerSocket serverSocket = new ServerSocket(PORT)) { System.out.println("Server listening on port " + PORT); while (true) { Socket clientSocket = serverSocket.accept(); System.out.println("New client connected: " + clientSocket); ClientHandler clientHandler = new ClientHandler(clientSocket, clients); clients.add(clientHandler); new Thread(clientHandler).start(); } } catch (IOException e) { e.printStackTrace(); } } } Client Side (ChatClient.java): import java.io.*; import java.net.*; import java.util.Scanner; public class ChatClient { public static void main(String[] args) { try { Socket socket = new Socket("localhost", 5555); System.out.println("Connected to server: " + socket); new Thread(new ServerListener(socket)).start(); new Thread(new ClientSender(socket)).start(); } catch (IOException e) { e.printStackTrace(); } } } This project includes a ClientHandler class for server-client communication, ServerListener class for listening to server messages, and ClientSender class for sending messages from the client.Controller Class (ItemController.java): import org.springframework.beans.factory.annotation.Autowired; import org.springframework.http.HttpStatus; import org.springframework.http.ResponseEntity; import org.springframework.web.bind.annotation.*; import javax.validation.Valid; import java.util.List; @RestController @RequestMapping("/api/items") public class ItemController { @Autowired private ItemService itemService; @GetMapping public List getAllItems() { return itemService.getAllItems(); } @PostMapping("/create") public ResponseEntity createItem(@Valid @RequestBody Item item) { Item createdItem = itemService.createItem(item); return new ResponseEntity(createdItem, HttpStatus.CREATED); } // Add additional CRUD operations as needed } Service Class (ItemService.java): import org.springframework.beans.factory.annotation.Autowired; import org.springframework.stereotype.Service; import java.util.List; @Service public class ItemService { @Autowired private ItemRepository itemRepository; public List getAllItems() { return itemRepository.findAll(); } public Item createItem(Item item) { // Add business logic for creating an item return itemRepository.save(item); } // Add additional service methods as needed } This project uses Spring Boot, with a controller handling HTTP requests, a service layer for business logic, and a repository for database interactions.WebScraper Class (WebScraper.java): import java.util.concurrent.ExecutorService; import java.util.concurrent.Executors; public class WebScraper { private static final int THREAD_POOL_SIZE = 5; private static final ExecutorService executorService = Executors.newFixedThreadPool(THREAD_POOL_SIZE); public static void main(String[] args) { // Add logic for dynamic URL queue, error handling, and data storage // Use executorService to manage multithreading } // Additional methods for fetching, parsing, and storing data } This project utilizes ExecutorService for managing a pool of threads to fetch and process web pages concurrently.MusicPlayer Class (MusicPlayer.java): import javafx.application.Application; import javafx.scene.Scene; import javafx.scene.control.Button; import javafx.scene.layout.VBox; import javafx.stage.Stage; public class MusicPlayer extends Application { public static void main(String[] args) { launch(args); } @Override public void start(Stage primaryStage) { primaryStage.setTitle("Music Player"); Button playButton = new Button("Play"); Button stopButton = new Button("Stop"); // Add more buttons for controls VBox vbox = new VBox(playButton, stopButton); // Add additional UI components for playlist, equalizer, etc. primaryStage.setScene(new Scene(vbox, 300, 200)); primaryStage.show(); } // Additional methods for handling audio playback, playlist, equalizer, etc. } This JavaFX Music Player project provides a basic user interface with buttons for play, stop, and can be expanded with additional controls.StockTradingSystem Class (StockTradingSystem.java): import java.util.List; import java.util.concurrent.Executors; import java.util.concurrent.ScheduledExecutorService; import java.util.concurrent.TimeUnit; public class StockTradingSystem { private static final ScheduledExecutorService scheduler = Executors.newScheduledThreadPool(1); public static void main(String[] args) { // Add logic for real-time market data, portfolio analytics, and algorithmic trading support // Schedule fetching real-time market data every minute scheduler.scheduleAtFixedRate(StockDataFetcher::fetchStockData, 0, 1, TimeUnit.MINUTES); } // Additional methods for portfolio analytics, algorithmic trading, etc. } This project uses a ScheduledExecutorService to fetch real-time market data at regular intervals and provides a foundation for portfolio analytics and algorithmic trading.These code snippets provide a detailed view of the structure and features of those 5 Java projects. Keep in mind that these are still simplified examples, and you may need to adapt and extend them based on specific project requirements and technologies used.In conclusion, diving into Java programming through these projects can be an exciting journey. Whether you’re a beginner taking your first steps or an experienced developer looking to enhance your skills, these projects offer a range of opportunities to learn and grow.For beginners, starting with projects like “Hello World GUI” and the “Number Guessing Game” provides a solid foundation. These projects introduce you to essential concepts like graphical user interfaces and basic programming logic in a fun and interactive way.As you progress, tackling projects like the “RESTful API with Spring Boot” and the “Multithreaded Web Scraper” opens doors to the world of web development and concurrent programming. These challenges enhance your skills and prepare you for real-world scenarios, especially if you’re aiming to build scalable and efficient applications.For the more advanced developers, taking on projects like the “JavaFX Music Player” and the “Stock Trading System” presents exciting opportunities. You’ll delve into multimedia development, user interface responsiveness, and even explore the intricate world of financial systems.Remember, the key is not just to complete these projects but to understand the underlying principles and continuously challenge yourself. Building a diverse portfolio of Java projects will not only boost your confidence but also make you a more versatile and capable developer.So, grab your coding tools, embark on these projects, and enjoy the rewarding experience of bringing your Java skills to new heights. Happy coding! If you’re looking to build a career as a Java developer and haven’t taken the plunge yet, this is the perfect place to start! You will work on a range of exciting projects in Java including a complete web framework, a REST API, a mobile app, and a distributed data processing application. We’ll provide you with the latest technology, and we’ll give you the chance to get stuck in and learn a lot at the same time! Here is a compilation of all the projects in Java and mini projects published in this site. All the projects are available with source code for free download! The projects listed here are mostly advanced projects developed using Java and many of these, but not all, use Oracle 10g database These can be downloaded in Eclipse, Netbeans, and Myeclipse IDEs. Nowadays, a lot of java projects – applications and software are developed in core Java, JSP, servlet, struts, spring, and hibernate technology. The ones in the following listing are developed using these technologies based on the type and scope of the project. All these projects can be used as semester or graduation projects by CSE and B.Tech final year students. The list below includes Projects in Java final year projects as well as mini projects built as either simple applications, big web-applications, or software. Almost all these Java projects have source code and database in the download file. In addition to source code and database, many projects are also available with project documentation, report, and paper presentation (PPT). Projects in Java available for download with project reports and documentations have SRS including objectives and scopes of the project, users of the proposed system and their role, system design explanation, functional and non-functional requirements, etc. Also, you can find system design diagrams and project screenshots in the documentation. We’ve also included project manuals in many projects; this document will guide you to run the project. For projects without documentation and report, you can refer the project introduction in respective post of each project as project synopsis. Only a few projects in this listing are without source code – only Java project report, and these are presented here just for the sake of reference. The Airline Reservation System java project is a program that assists with checking flights. The application is intended to be used by both commercial and private users. This project consists of 4 parts, each with their own responsibilities. The user interface is responsible for handling requests from the client side while the server is responsible for storing the data and running the database queries. The third part contains the database, the fourth part is responsible for logging the user data. Bug tracking system java project with source code can be helpful to companies to be successful. It is necessary to manage the bug reports to avoid bugs. With the help of bug tracking system java project with source code, companies can avoid many errors and bugs. It is also possible to fix these bugs quickly and avoid many inconveniences. Bug Tracking System Java with Source Code is a software which is written using the Java programming language and developed under open source licenses. The aim of this project is to build a library which can be used to track bugs in the Java projects. Career Information Management System java project with source code. You may have heard of a career information system, but do you know exactly what it does and how to build one? Most career information systems are built by IT departments for internal use, but many companies are starting to build these systems so that they can be used externally by HR professionals. The best career information systems integrate multiple data sources into one central database where you can perform searches, view reports, create profiles, and compare jobs. You can read more about career information systems on our blog page. Course management system, also known as CRS or CMS, is software which enables educators to manage content, deliver courses, and distribute exams. In recent years, it has gained widespread use in higher education and is becoming increasingly popular in K-12 schools. Most CRS software packages contain a number of features, including course scheduling, lesson design, student tracking, grade book, and course report distribution. Course Management System java project with source code (CMS Java project with source code) is about the course management system project, it’s is a project based on the source code of course management system in java, which provides a platform for the management of the educational content in different educational levels. Currency Converter Java Project is a Java project based on Java SE API for performing conversion between any two currencies. This project is also known as Currency Converter and Exchange Rate Calculator. Currency converter is one of the basic applications, which can be used in our daily life. Currency converter is the application which converts currency between different countries. Currency converter is used when we travel abroad, while visiting foreign lands. In order to get started, let’s talk about the currency converter program that we’ll use for our example. We will use a web service that allows us to convert currencies online. You can find a list of currency conversion websites. Data Visualization Software Java Project With Source Code. As the amount of data being generated by data sources increases exponentially each year, organizations are turning to data visualization software to make sense of their vast amounts of data. The idea behind this project is to design a software which will analyze the data and provide information about it in graphical representation using simple charts. This is the Java Steganography project where we will learn what is digital steganography java project with source code. It’s a very simple, but yet a very powerful application which helps to hide one picture inside another. It is used to hide messages in a image. Steganography is the practice of concealing information in a way that makes it undetectable, so you can then send that information securely without anyone being able to access it. Steganography can be used for secret communication, hiding information from spies and censors, hiding money, or even hiding data about an individual from the authorities. With steganography, it’s possible to transmit a secret message by embedding it into a seemingly harmless picture. E-Health Care Management System java project with source code. In E health care management system, the health care professionals can use this system to manage their patients. It gives the patient access to the health care services which helps in proper management of their health care activities. The E health care management system java project with source code provides the services such as managing and organizing the patients’ medical files, providing online appointment scheduling, etc. Electricity Billing System java project with source code. The project is to develop an electricity billing system. The data entered in the system must be displayed in a list. There must be a graphical representation of the billing cycles. This will include the days of the month, the number of units billed per day, the price per unit and the total price of each bill. Each unit is charged based on the price at which it is purchased (the price per unit). When the total of all bills is zero, the user must be informed about it. An Email Client software in Java is an email program developed to send and receive electronic mail. This proposed project is an Email Client application developed using Java and some Java APIs. Here, standard sockets and networking have been used in addition to Java Mail API. The protocols used in the project are SMTP, POP3; the latter one is used to collect the emails sent to our address. In the event management system (EMS) java project, we would like to make sure that the user is able to submit the information for the event they are organizing as well as retrieve the information about the previous events organized by the user. This application would include the ability to add the event details, including date, time, place, organization, etc. as well as the option to delete or modify the details of the event. Also, the event organizer would be able to see the details of the previous events organized by him or her. This application would be used to help the organizers of the events organize them efficiently and effectively. The Farmers Management System is a comprehensive management system for farmers designed to help them manage every aspect of their farm efficiently and effectively. Farmers Buddy is a open discussion portal developed using Java programming language with Oracle as the system database. This project is useful for farmers and agricultural students to obtain information regarding various crops, the fertilizers usage for these crops, the soil condition in which these crops yield more, and the suitable climatic and environmental condition for these crops. In this tutorial, you’ll learn how to create an online chatting application with file transfer functionality using Java. You will get all the source code of this project with step by step instructions. We’ll be working on FileTransferChatClient Java application, where client sends files to a server and server returns chat messages back to client. FileTransferClientServer is a Java Server-side application running on the server. The Client side application, FileTransferChatClient Java, is responsible for sending and receiving files to and from the server. The chat server is written in Java, using the Socket programming API. It’s a simple Java application that allows you to evaluate the performance of your employees. You can use the Java project source code to make your own custom Java application. In Library Management System you need to handle several important tasks. One of them is a searching in different library catalogues which will give you a result with a lot of information about books. Another one is that it’s possible to add book with it’s title, author, publisher and so on. The third task is the searching for some particular book, which will be found very quickly. But we have the biggest task – downloading the whole file. It means that we need to download the whole file from the internet. It’s very important that we don’t spend too much time on this downloading and the speed of the download is quite important. You may be looking for a way to start with your own Java mail server. Well, it is possible to start from scratch. You don’t have to do it by yourself and start learning from scratch. There are already pre-built versions of Java mail server system available. If you know where to look, then you will come across several pre-built Java mail servers. But, if you are a complete beginner, then you should opt for a ready made one. This will save you from a lot of trouble. Mail Server System is an interesting and complex topic. When you think about it, the Internet is nothing more than a giant network of computers that store information and communicate using a set of protocols and standards called TCP/IP. TCP/IP is used by all the computers connected to the Internet to send data back and forth between each other. As a result, it becomes incredibly important for these computers to be able to talk to one another, and this requires a protocol called IP, which stands for Internet Protocol. One of the main tasks of a mail server is to handle all of this communication between the Internet’s computers. This includes accepting emails that arrive from outside of the network, filtering out spam, storing email messages in its inbox, and delivering those messages to the appropriate recipients. This project shows how to use Java and Packet Sniffing to detect the traffic between two hosts. This application allows user to sniff packets and track the TCP/IP connections of the host in real-time. Packet Sniffing is a technique of intercepting the packets of the data communication across the network between two hosts. As we know that, TCP/IP is the main protocol used by computers to communicate. When the packets are intercepted, the user can see the source IP address and destination IP address of the packets. These addresses can be seen in the table when you click on the link. This Number Guessing Game Java Project is based on the famous game where users are asked to guess what number is being hidden. You will learn how to create a number guessing game in this Java tutorial. If you are familiar with this Java project To develop an Online Attendance Management System, we use Java as our programming language, MySQL Database and JSP (JavaServer Pages) for server side programming, XML and HTML for client side programming. An attendance management system is a solution that is offered by organizations so that they can manage their employees. It is a software application that helps you track the attendance of your staff. These applications can also help you manage the leaves of your staff. By doing so, you are able to prevent any kind of mistake in your attendance management system. You can also track the absence of your staff, you can monitor the performance of the staff, and you can generate reports regarding the attendance of the staff. In this article, we’ll discuss about the basic definition of an attendance management system and the reasons why you need one. The Online Bank Management System java project with source code is designed to help you manage and maintain your bank accounts online. It has a web-based interface to help you with your banking activities and manage your account information. It mainly aims at making the banking system easily accessible from anywhere and improving and enhancing the running banking process. With the use of this online software, bank customers don’t need to go to bank to make inquiry regarding their balance or to transfer their balance to other account or for any other banking services. Online Book Store Online Customer Care and Service Center Online Document Management System Online Examination Management System Online Medical Management System Online Quiz System Online Survey System Photo Lab Management System RSS Feed Reader Smart City Project Stock Management System Student Result Processing System Supply Chain Management System Telephone Billing System Travel Management System University Admission Management System Virtual Private Network Web Server Management System In addition to the topics aforementioned, here are some interesting project ideas and topics for final year students who’re looking forward to doing their graduation project in Java. Some of these topics would suit simple java programs, games, and mini projects as well. Choose the topic based on the scope and nature of your project. Address Book Management System Affiliate Manager Ajax Browser Application Installer Software Application Re-installer Software ATM Simulator System Automatic File Update Bookmark Sync Bus Ticket Reservation System Calendar Icon Maker Application CSS Color and Image Annotator Cybercafe Mangement System Data Encryption Database Explorer Directory and File Explorer Domain Search Engine E-Acquisition E-Advertisement E-Learning Enterprise Scheme Planner Face Identification File and Folder Explorer Font Detector/Finder HTML Color Code Finder HTML Spell Checker Image Compressor Application Image to Text Converter Internet Credit Card System Internet Usage Monitoring System Intrusion Detection in Wireless Sensor Network Keyword Finder and Number Calculator Language Emulator Mac Ethernet Address Malware Scanner MP4 Video Converter Online Auction Online Magazine Password Protector and Reminder PDF Converter Pharmacy Management System Port Scanner Process Analysis in Asynchronous System Remote Desktop Administrator Resource Planner and Organizer Search Engine Security System for DNS using Cryptography SmartFTP Uploads Synchronous Conferencing System Text to HTML Converter Unicode Font Voice Chatting and Video Conferencing Voice Compressor Software Windows Fixer XML Compactor People often ask us “Which project/topic should I pick?”. Well, it’s entirely up to you. Here’s what we’d like to suggest: pick several topics and narrow down your choices into a bunch based on your level of knowledge and level of project difficulty. Free Projects in Java are usually created to solve a problem or fulfill a need. The idea is to share your knowledge and help others learn new things in an organized way. This is a great way to get involved in Open Source and learn a new language, programming framework or tool that could be helpful for your future career. Check out this list of projects to find out more. Then, narrow download these topics based on several things such as whether they include servers, how complex they may get as you go on coding, etc. Eliminate the topics/projects that seem boring to you. Overall, choose the project or topic based on the nature and scope of the application you’ve been assigned or you’re going to develop. Similar lists: ASP.NET Projects C/C++ Projects PHP Projects These are all the projects tagged “Java” available in this site. We add new projects on a regular basis, so you can bookmark this page and stay updated with the latest Java projects with their source code and project documents. If you have a project request or a smart project idea that you’d like to see in our listing, don’t hesitate to drop a mail at codewithc2014@gmail.com or you can submit your queries from the comments section. You can’t perform that action at this time.