Explore More

Subcription : Premium CDAC NOTES & MATERIAL @99

Contact to Join
Premium Group

Click to Join
Telegram Group

For More E-Notes

Join Our Community to stay Updated

TAP ON THE ICONS TO JOIN!

	codewitharrays.in freelance project available to buy contact on 8007592194	
SR.NO	Project NAME	Technology
1	Online E-Learning Platform Hub	React+Springboot+MySql
2	PG Mates / RoomSharing / Flat Mates	React+Springboot+MySql
3	Tour and Travel management System	React+Springboot+MySql
4	Election commition of India (online Voting System)	React+Springboot+MySql
5	HomeRental Booking System	React+Springboot+MySql
6	Event Management System	React+Springboot+MySql
7	Hotel Management System	React+Springboot+MySql
8	Agriculture web Project	React+Springboot+MySql
9	AirLine Reservation System / Flight booking System	React+Springboot+MySql
10	E-commerce web Project	React+Springboot+MySql
11	Hospital Management System	React+Springboot+MySql
12	E-RTO Driving licence portal	React+Springboot+MySql
13	Transpotation Services portal	React+Springboot+MySql
14	Courier Services Portal / Courier Management System	React+Springboot+MySql
15	Online Food Delivery Portal	React+Springboot+MySql
16	Muncipal Corporation Management	React+Springboot+MySql
17	Gym Management System	React+Springboot+MySql
18	Bike/Car ental System Portal	React+Springboot+MySql
19	CharityDonation web project	React+Springboot+MySql
20	Movie Booking System	React+Springboot+MySql

	freelance_Project available to buy contact on 8007592194	
21	Job Portal web project	React+Springboot+MySql
22	LIC Insurance Portal	React+Springboot+MySql
23	Employee Management System	React+Springboot+MySql
24	Payroll Management System	React+Springboot+MySql
25	RealEstate Property Project	React+Springboot+MySql
26	Marriage Hall Booking Project	React+Springboot+MySql
27	Online Student Management portal	React+Springboot+MySql
28	Resturant management System	React+Springboot+MySql
29	Solar Management Project	React+Springboot+MySql
30	OneStepService LinkLabourContractor	React+Springboot+MySql
31	Vehical Service Center Portal	React+Springboot+MySql
32	E-wallet Banking Project	React+Springboot+MySql
33	Blogg Application Project	React+Springboot+MySql
34	Car Parking booking Project	React+Springboot+MySql
35	OLA Cab Booking Portal	React+NextJs+Springboot+MySql
36	Society management Portal	React+Springboot+MySql
37	E-College Portal	React+Springboot+MySql
38	FoodWaste Management Donate System	React+Springboot+MySql
39	Sports Ground Booking	React+Springboot+MySql
40	BloodBank mangement System	React+Springboot+MySql

41	Bus Tickit Booking Project	React+Springboot+MySql
42	Fruite Delivery Project	React+Springboot+MySql
43	Woodworks Bed Shop	React+Springboot+MySql
44	Online Dairy Product sell Project	React+Springboot+MySql
45	Online E-Pharma medicine sell Project	React+Springboot+MySql
46	FarmerMarketplace Web Project	React+Springboot+MySql
47	Online Cloth Store Project	React+Springboot+MySql
48	Train Ticket Booking Project	React+Springboot+MySql
49	Quizz Application Project	JSP+Springboot+MySql
50	Hotel Room Booking Project	React+Springboot+MySql
51	Online Crime Reporting Portal Project	React+Springboot+MySql
52	Online Child Adoption Portal Project	React+Springboot+MySql
53	online Pizza Delivery System Project	React+Springboot+MySql
54	Online Social Complaint Portal Project	React+Springboot+MySql
55	Electric Vehical management system Project	React+Springboot+MySql
56	Online mess / Tiffin management System Project	React+Springboot+MySql
57		React+Springboot+MySql
58		React+Springboot+MySql
59		React+Springboot+MySql
60		React+Springboot+MySql

Spring Boot + React JS + MySQL Project List

Sr.No	Project Name	YouTube Link
1	Online E-Learning Hub Platform Project	https://youtu.be/KMjyBaWmgzg?si=YckHuNzs7eC84-IW
2	PG Mate / Room sharing/Flat sharing	https://youtu.be/4P9clHg3wvk?si=4uEsi0962CG6Xodp
3	Tour and Travel System Project Version 1.0	https://youtu.be/-UHOBywHaP8?si=KHHfE_A0uv725f12
4	Marriage Hall Booking	https://youtu.be/VXz0kZQi5to?si=llOS-QG3TpAFP5k7
5	Ecommerce Shopping project	https://youtu.be/vJ_C6LkhrZ0?si=YhcBylSErvdn7paq
6	Bike Rental System Project	https://youtu.be/FIzsAmIBCbk?si=7ujQTJqEgkQ8ju2H
7	Multi-Restaurant management system	https://youtu.be/pvV-pM2Jf3s?si=PgvnT-yFc8ktrDxB
8	Hospital management system Project	https://youtu.be/IynIouBZvY4?si=CXzQs3BsRkjKhZCw
9	Municipal Corporation system Project	https://youtu.be/cVMx9NVyI4I?si=qX0oQt-GT-LR_5jF
10	Tour and Travel System Project version 2.0	https://youtu.be/_4u0mB9mHXE?si=gDiAhKBowi2gNUKZ

Sr.No	Project Name	YouTube Link
11	Tour and Travel System Project version 3.0	https://youtu.be/Dm7nOdpasWg?si=P_Lh2gcOFhlyudug
12	Gym Management system Project	https://youtu.be/J8_7Zrkg7ag?si=LcxV51ynfUB7OptX
13	Online Driving License system Project	https://youtu.be/3yRzsMs8TLE?si=JRI_z4FDx4Gmt7fn
14	Online Flight Booking system Project	https://youtu.be/m755rOwdk8U?si=HURvAY2VnizIyJlh
15	Employee management system project	https://youtu.be/lD1iE3W_GRw?si=Y_jv1xV_BljhrD0H
16	Online student school or college portal	https://youtu.be/4A25aEKfei0?si=RoVgZtxMk9TPdQvD
17	Online movie booking system project	https://youtu.be/Lfjv_U74SC4?si=fiDvrhhrjb4KSlSm
18	Online Pizza Delivery system project	https://youtu.be/Tp3izreZ458?si=8eWAOzA8SVdNwlyM
19	Online Crime Reporting system Project	https://youtu.be/0UlzReSk9tQ?si=6vN0e70TVY1GOwPO
20	Online Children Adoption Project	https://youtu.be/3T5HC2HKyT4?si=bntP78niYH802I7N

✓ **What will the output of the code be?** * 1/1

```
public class PrintTest {
public static void main(String[] args) {
System.out.print("Hello ");
System.out.println("World!");
System.out.printf("Number: %d", 10);
}
}
```

- ○ Hello World!Number: 10
- ○ Hello World! Number: 10
- ◉ Hello World! /n Number: 10 ✓
- ○ HelloWorld!Number: 10

✗ **What is the significance of using String... args instead of String[] args in the main method?** *0/1

- ○ It is an invalid syntax.
- ◉ It allows passing multiple string arguments in the command line. ✗
- ○ It does not affect functionality; both are equivalent.
- ○ It prevents passing arguments to the program.

Correct answer

- ◉ It does not affect functionality; both are equivalent.

What concept is demonstrated in Line 1? *

0/1

```
public class Test {
public static void main(String[] args) {
int a = 10;
Integer b = a; // Line 1
System.out.println(b);
}
}
```

- (●) Implicit Unboxing ✗
- ( ) Explicit Boxing
- ( ) Implicit Boxing
- ( ) Explicit Unboxing

Correct answer

- (●) Implicit Boxing

What concept is demonstrated in Line 2? *

0/1

```
public class Test {
public static void main(String[] args) {
Integer a = 15;
int b = a; // Line 2
System.out.println(b);
}
}
```

- ( ) Explicit Boxing
- ( ) Implicit Unboxing
- (•) Implicit Boxing ✗
- ( ) Explicit Unboxing

Correct answer

- (•) Implicit Unboxing

✓ **What will happen when the code at Line 1 is executed?** * 1/1

```
public class Test {
public static void main(String[] args) {
String str = "abc";
int num = Integer.parseInt(str); // Line 1
System.out.println(num);
}
}
```

- ○ It will compile and print abc.
- ○ It will compile and print 0.
- ◉ It will throw a NumberFormatException. ✓
- ○ It will throw a NullPointerException.

✓ **What will happen when the code at Line 1 is executed?** * 1/1

```
public class Test {
public static void main(String[] args) {
String[] arr = new String[3];
arr[0] = "Java";
System.out.println(arr[1].toUpperCase()); // Line 1
}
}
```

- ○ It will compile and print null.
- ○ It will compile and print JAVA.
- ○ It will throw an ArrayIndexOutOfBoundsException.
- ◉ It will throw a NullPointerException. ✓

✓ **Which of the following is a correct example of Widening Conversion in Java?** *1/1

- ○ int i = 10; byte b = i;
- ○ double d = 10.5; int i = d;
- ◉ float f = 10; double d = f; ✓
- ○ long l = 100; int i = l;

✓ **Which of the following requires an explicit cast for Narrowing Conversion in Java?** *1/1

- (●) double d = 100.25; int i = (int) d; ✓
- ( ) int i = 50; long l = i;
- ( ) byte b = 100; int i = b;
- ( ) loat f = 10.5F; double d = f;

✓ **Which of the following statements is true about the memory storage of a and b in the given code?** *1/1

```
public class Test {
public static void main(String[] args) {
int a = 10; // Line 1
String b = "Hello"; // Line 2
}
}
```

- ( ) Both a and b are stored in the heap memory.
- (●) a is stored in the stack memory, while b is stored in the heap memory. ✓
- ( ) Both a and b are stored in the stack memory.
- ( ) a is stored in the heap memory, while b is stored in the stack memory.

✓ **What are the default values of primitive and non-primitive data types in Java?** *1/1

- ○ Primitive types have default values of null, and non-primitive types have default values of 0.
- ◉ Primitive types have default values based on their type (e.g., 0 for int, false for boolean), and non-primitive types have null as their default value. ✓
- ○ Both primitive and non-primitive types have null as their default value.
- ○ Both primitive and non-primitive types have 0 as their default value.

✗ **Which of the following static methods is common to all wrapper classes in Java (such as Integer, Double, and Character)?** *0/1

- ○ parseInt(String s)
- ○ valueOf(String s)
- ◉ toString() ✗
- ○ compareTo(T another)

Correct answer

- ◉ valueOf(String s)

✓ **What will be the output of this code?** * 1/1

```
public class Test {
public static void main(String[] args) {
double d = 9.78;
int i = (int) d; // Line 1
System.out.println(i);
}
}
```

- (●) 9 ✓
- ( ) 9.78
- ( ) 10
- ( ) Error

✓ **Consider the following Java code:** *1/1

```
public class BankAccount {
static double interestRate = 0.03;

static void updateInterestRate(double newRate) {
interestRate = newRate;
}

double balance;

void deposit(double amount) {
if (amount > 0) {
balance += amount;
}
}
}

public class Main {
public static void main(String[] args) {
BankAccount.updateInterestRate(0.05);

BankAccount account = new BankAccount();
account.deposit(500.00);

System.out.println("Interest Rate: " + BankAccount.interestRate);
```

```
System.out.println("Account Balance: " + account.balance);

}

}
```

Which of the following statements is correct regarding the code execution?

- updateInterestRate can be called on the BankAccount instance, and deposit can be called on the class BankAccount.
- updateInterestRate can be called directly on the BankAccount class, and deposit must be called on an instance of BankAccount. ✓
- updateInterestRate can only be called on an instance of BankAccount, and deposit can be called on the BankAccount class.
- Both updateInterestRate and deposit can be called directly on the BankAccount class.

Given the following Java class: *1/1

```
public class Customer {

String customerName;

double accountBalance;


void deposit(double amount) {

if (amount > 0) {

accountBalance += amount;

}

}


static void setDefaultBalance(double defaultBalance) {

// This method should set a default balance for all customers

}

}
```

Which of the following statements is correct about customerName, accountBalance, and setDefaultBalance?

- customerName and accountBalance are static variables; setDefaultBalance is a non-static method.
- customerName and accountBalance are instance variables; setDefaultBalance is a static method. ✓
- customerName is a static variable, accountBalance is a non-static variable, and setDefaultBalance is an instance method.
- Both customerName and accountBalance are static variables; setDefaultBalance is an instance method.

Consider the following Java method and its invocation: *0/1

```
public class Calculator {
void addNumbers(int num1, int num2) {
System.out.println("Sum: " + (num1 + num2));
}

public static void main(String[] args) {
Calculator calc = new Calculator();
calc.addNumbers(10, 20);
}
}
```

Which of the following statements correctly describes the terms "parameters" and "arguments" in the context of the provided code?

- (●) num1 and num2 are arguments, and 10 and 20 are parameters. ✗
- ( ) 10 and 20 are parameters, and num1 and num2 are arguments.
- ( ) num1 and num2 are parameters, and 10 and 20 are arguments.
- ( ) Both num1 and num2, as well as 10 and 20, are parameters.

Correct answer

- (●) num1 and num2 are parameters, and 10 and 20 are arguments.

✓ **Given the following code snippet:** * 1/1

```
public class Test {
public static void main(String[] args) {
System.out.print("Hello, ");
System.out.print("World!");
}
}
```

What is the role of out in this context?

- (●) out is an instance of the PrintStream class used for printing output to the console. ✓
- ( ) out is a method that formats the output before printing it to the console.
- ( ) out is a variable that stores the current state of the system.
- ( ) out is a class that handles file operations in Java.

✓ 1. The JVM divides memory into different regions such as the Heap, Stack, and Method Area. *1/1

2. The Garbage Collector (GC) primarily manages the Stack memory.

3. The Method Area stores class metadata and static variables.

Which of the following statements is correct?

- (●) Only statements 1 and 3 are correct; the Garbage Collector manages the Heap memory, not the Stack. ✓
- ( ) All statements are correct.
- ( ) Only statement 1 is correct; the Garbage Collector does not manage the Method Area.
- ( ) Only statement 3 is correct; the Stack and Heap memory are not managed by the Garbage Collector.

✓ **Which of the following accurately describes the role of the JVM Execution Engine?** *1/1

- It compiles Java bytecode into native machine code for execution on the host system.
- It translates Java source code into bytecode, which is then executed by the Java Compiler.
- ◉ It interprets or compiles Java bytecode into native machine code for execution, and manages runtime optimizations such as Just-In-Time (JIT) compilation. ✓
- It handles network communication and database interactions during Java application execution.

✓ **Which of the following statements about Java data types is correct?** * 1/1

- The float data type has a higher precision than the double data type.
- ◉ char can hold any Unicode character and is stored as a 16-bit integer. ✓
- The boolean data type can store multiple values like true, false, and null.
- The long data type is used to store decimal numbers with higher precision than float.

✓ **Which of the following option leads to the portability and security of Java?** *1/1

- ◉ Bytecode is executed by JVM ✓
- The applet makes the Java code secure and portable
- Use of exception handling
- Dynamic binding between objects

Feedback of Mock 0 of 0 points

https://www.youtube.com/@codewitharrays

https://www.instagram.com/codewitharrays/

https://t.me/codewitharrays Group Link: https://t.me/ccee2025notes

+91 8007592194 +91 9284926333

codewitharrays@gmail.com

https://codewitharrays.in/project