Java exam. topics Data types Oop concepts Inheritance and types of inheritance Awt and applet Interfaces String handling Constructors and types Exception handling Threads sample program for every question even it is not fully correct.

some sample questions and programs for each topic:

1. Data types:

Q: What is the difference between int and double data types in Java?

A: Int is used for storing integer values, while double is used for storing decimal values.

Sample program:

public class Main {

public static void main(String[] args) {

int num1 = 10;

double num2 = 3.14;

System.out.println("num1 = " + num1);

System.out.println("num2 = " + num2);

}

}

1. OOP concepts:

Q: What is encapsulation in Java?

A: Encapsulation is the process of wrapping code and data together into a single unit, and making sure that the data is only accessible through methods defined in the class.

Sample program:

public class Person {

private String name;

private int age;

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

}

public class Main {

public static void main(String[] args) {

Person person = new Person();

person.setName("John");

person.setAge(30);

System.out.println("Name: " + person.getName());

System.out.println("Age: " + person.getAge());

}

}

1. Inheritance and types of inheritance:

Q: What is the difference between single inheritance and multiple inheritance?

A: Single inheritance is when a subclass inherits from a single superclass, while multiple inheritance is when a subclass inherits from multiple superclasses.

Sample program:

public class Animal {

public void eat() {

System.out.println("Eating...");

}

}

public class Dog extends Animal {

public void bark() {

System.out.println("Barking...");

}

}

public class Cat extends Animal {

public void meow() {

System.out.println("Meowing...");

}

}

public class Main {

public static void main(String[] args) {

Dog dog = new Dog();

dog.eat();

dog.bark();

Cat cat = new Cat();

cat.eat();

cat.meow();

}

}

1. AWT and Applet:

Q: What is an Applet in Java?

A: An applet is a small program that runs within a web browser. It can be used to create interactive animations, games, and other applications.

Sample program:

import java.awt.\*;

import java.applet.\*;

public class MyApplet extends Applet {

public void paint(Graphics g) {

g.drawString("Hello, world!", 50, 50);

}

}

// HTML code to embed the applet in a web page

<applet code="MyApplet.class" width="300" height="200"></applet>

1. Interfaces:

Q: What is an interface in Java?

A: An interface is a collection of abstract methods that can be implemented by any class that implements the interface.

Sample program:

{ this.length = length; this.width = width; }

public double getArea() { return length \* width; }

public double getPerimeter() { return 2 \* (length + width); } }

public class Main { public static void main(String[] args) { Circle circle = new Circle(5); System.out.println("Circle area: " + circle.getArea()); System.out.println("Circle perimeter: " + circle.getPerimeter());

Rectangle rectangle = new Rectangle(10, 5);

System.out.println("Rectangle area: " + rectangle.getArea());

System.out.println("Rectangle perimeter: " + rectangle.getPerimeter());

} }

v

6. String handling:

Q: What is the difference between String and StringBuffer in Java?

A: String is an immutable class, while StringBuffer is a mutable class.

Sample program:

public class Main { public static void main(String[] args) { String str1 = "Hello"; str1 += " world!"; System.out.println("String: " + str1);

StringBuffer str2 = new StringBuffer("Hello");

str2.append(" world!");

System.out.println("StringBuffer: " + str2);

} }

7. Constructors and types:

Q: What is a constructor in Java?

A: A constructor is a special method that is called when an object is created. It is used to initialize the object's data members.

Sample program:

public class Person { private String name; private int age;

public Person(String name, int age) { this.name = name; this.age = age; }

public String getName() { return name; }

public int getAge() { return age; } }

public class Main { public static void main(String[] args) { Person person = new Person("John", 30); System.out.println("Name: " + person.getName()); System.out.println("Age: " + person.getAge()); } }

8. Exception handling:

Q: What is exception handling in Java?

A: Exception handling is the process of handling errors and exceptions that occur during the execution of a program. It is used to ensure that the program continues to run even if an error occurs.

Sample program:

public class Main { public static void main(String[] args) { try { int num1 = Integer.parseInt(args[0]); int num2 = Integer.parseInt(args[1]); int result = num1 / num2; System.out.println("Result: " + result); } catch (NumberFormatException e) { System.out.println("Invalid input!"); } catch (ArithmeticException e) { System.out.println("Cannot divide by zero!"); } } }

9. Threads:

Q: What is a thread in Java?

A: A thread is a separate path of execution within a program. It allows multiple tasks to be performed simultaneously.

Sample program:

public class MyThread extends Thread { public void run() { for (int i = 1; i <= 5; i++) { System.out.println("Thread " + Thread.currentThread().getId() + ": " + i); } } }

public class Main { public static void main(String[] args) { MyThread thread1 = new MyThread(); MyThread thread2 = new MyThread(); thread1.start(); thread2.start(); } }

I hope these examples help you in your Java exam preparation!