# Questions

1. Write a Java program that demonstrates the use of inheritance. Create a base class called "Person" with two properties: name and age. Then create a subclass called "Employee" that inherits from the "Person" class and adds a new property called "salary". Finally, create an instance of the "Employee" class and print out all of its properties. (10 marks)

2. Write a Java program that demonstrates the use of polymorphism. Create a base class called "Shape" with a method called "calculateArea()". Then create two subclasses: "Rectangle" and "Circle". Each subclass should override the "calculateArea()" method to calculate the area of the respective shape. Finally, create an instance of each subclass and call the "calculateArea()" method on each instance. (10 marks)

3. Write a Java program that demonstrates the use of encapsulation. Create a class called "BankAccount" with two private properties: "balance" and "accountNumber". Then create getter and setter methods for each property. Finally, create an instance of the "BankAccount" class and use the getter and setter methods to set and retrieve the values of the properties. (10 marks)

4. Write a Java program that demonstrates the use of abstraction. Create an abstract class called "Animal" with a method called "makeSound()". Then create two subclasses: "Dog" and "Cat". Each subclass should override the "makeSound()" method to make the respective animal sound. Finally, create an instance of each subclass and call the "makeSound()" method on each instance. (10 marks)

5. Write a Java program that demonstrates the use of interfaces. Create an interface called "Shape" with two methods: "calculateArea()" and "calculatePerimeter()". Then create two classes: "Rectangle" and "Circle". Each class should implement the "Shape" interface and provide implementations for the two methods. Finally, create an instance of each class and call the "calculateArea()" and "calculatePerimeter()" methods on each instance. (10 marks)

6. Write a Java program that demonstrates the use of static methods and variables. Create a class called "MathUtil" with a static method called "factorial()". The "factorial()" method should take an integer as input and return the factorial of that integer. Then create another class called "Main" and call the "factorial()" method on the "MathUtil" class. (10 marks)

7. Write a Java program that demonstrates the use of constructors. Create a class called "Person" with two properties: "name" and "age". Then create a constructor for the "Person" class that takes two parameters: "name" and "age". Finally, create an instance of the "Person" class using the constructor and print out its properties. (10 marks)

8. Write a Java program that demonstrates the use of exception handling. Create a method called "divide()" that takes two integers as input and returns the result of dividing the first integer by the second integer. If the second integer is zero, the method should throw an exception. Finally, call the "divide()" method with different inputs, including a second integer of zero, and handle any exceptions that are thrown. (10 marks)

9. Write a Java program that demonstrates the use of threads. Create a class called "Counter" that implements the "Runnable" interface. The "Counter" class should have a method called "run()" that counts from 1 to 10 and prints out each number. Then create two instances of the "Counter" class and start each instance in a separate thread. Finally, wait for both threads to finish before terminating the program. (10 marks)

10. Write a Java program that demonstrates the use of collections. Create an ArrayList called "students" and add five Student objects to it. Each Student object should have a name and an age property. Then use an Iterator to iterate over the "students" ArrayList and print out the name and age of each student. Finally, sort the "students" ArrayList by age in ascending order and print out the name and age of each student again. (10 marks)