Inheritance and Polymorphism

1. What is the difference between overloading and overriding a method?

Overloading: Two methods with the same name but different signatures in one class

Overriding: Two methods with the same signature, one in a superclass and one in a subclass

1. How many superclasses can a class have? 1
2. One class may be extended by many subclasses. (Choose: one, two or many)
3. Consider the following classes:

public class Car { ... }

public class ElectricCar extends Car { ... }

public class Tesla extends ElectricCar { ... }

Which of the following are legal statements? (Write yes or no next to each statement.)

1. Car a = new ElectricCar(); y
2. Car a = new Tesla(); y
3. ElectricCar b = new Tesla(); y
4. Tesla b = new Tesla(); n
5. Tesla b = new ElectricCar(); n
6. ElectricCar b = new Car(); n
7. Explain the difference between the this keyword and the super keyword. When should each be used?

This is used when referring to methods and variables of that instance, super is used when referring to methods or variables that belong to a superclass

1. Imagine that you are going to write a program to play card games. Consider a design with a Card class and 52 subclasses, one for each of the unique playing cards (for example, NineOfSpades and JackOfClubs). Is this a good design? If yes, why? If not, why not, and what might be a better design?

No, because that is too much code. One card class with suit and number fields is better.

1. The ability of the same code to be used with several different types of objects and for the code to behave differently depending on the actual type of object used is called Polymorphism.
2. Is it legal for a superclass variable to refer to an object of its subclass? no Given an example statement.
3. Give 3 situations where using a variable that is a superclass of other objects is beneficial. You can , \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, and \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,
4. When does a program choose which method to execute? (compile time, runtime, downtime etc.) runtime . This is called dynamic binding.
5. Given the statement Secretary sam = new LegalSecretary();, which of the following subsequent statements will cause a compiler error? b

a. sam.takeDictation(“Help me understand this code.”);

b. sam.fileLegalBriefs();

c. ((LegalSecretary) sam).takeDictation(“Help me understand this code.”);

d. ((LegalSecretary) sam).fileLegalBriefs();

e. (LegalSecretary) sam.takeDictation(“Help me understand this code.”);

f. (LegalSecretary) sam.fileLegalBriefs();

1. What is the output of the following code segment given the default vacation days is 10, a Lawyer receives 15 vacation days and a Secretary receives 12 vacation days?

10

15

12

Employee[] employees = {new Employee(), new Lawyer(), new Secretary()};

for (Employee e : employees) {

System.out.println(e.getVacationDays());

}

1. When evaluating complex code where polymorphism exists, first identify the top class in the hierarchy. What do you look for to find the top class in the hierarchy? The javadoc.
2. What is the output from the following code? MV2E1

public class E {

public void m1() { System.out.print(“E1”); }

public void m2() { System.out.print(“E2”); }

}

public class U extends MV {

public void m1() { System.out.print(“U1”); }

}

public class MV extends E {

public void m2() { System.out.print(“MV2”); }

}

public class Client {

public static void main(String[] args) {

E e = new E(); U u = new U(); MV mv = new MV();

u.m2();

mv.m1();

}

}