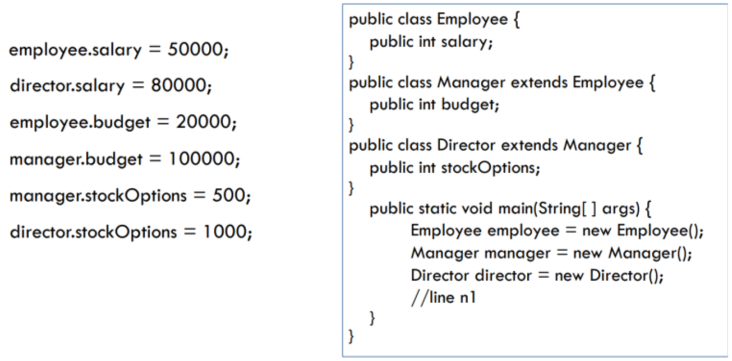
Vegas, Remie Ross E.

OOP 06 Quiz 1

1. **Select the four (4) statements that can be inserted at line n1.**



employee.salary = 50000;

director.salary = 80000;

employee.budget = 20000;

manager.budget = 100000;

manager.stockOptions = 500;

director.stockOptions = 1000;

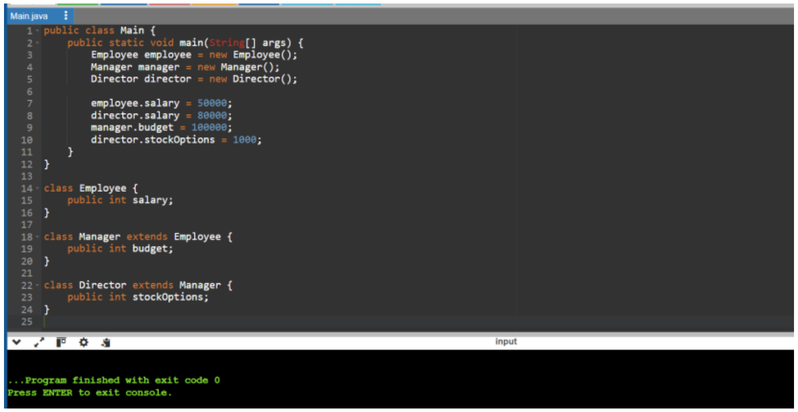
```
public class Employee {
    public int salary;
}
public class Manager extends Employee {
    public int budget;
}
public class Director extends Manager {
    public int stockOptions;
}
    public static void main(String[ ] args) {
            Employee employee = new Employee();
            Manager manager = new Manager();
            Director director = new Director();
            //line n1
    }
}
```

Answer:

The four valid statements are:

1. employee.salary = 50000;
2. director.salary = 80000;
3. manager.budget = 100000;
4. director.stockOptions = 1000;

Output:



```
public class Main {
    public static void main(String[] args) {
        Employee employee = new Employee();
        Manager manager = new Manager();
        Director director = new Director();

        employee.salary = 50000;
        director.salary = 80000;
        manager.budget = 100000;
        director.stockOptions = 1000;
    }
}

class Employee {
    public int salary;
}

class Manager extends Employee {
    public int budget;
}

class Director extends Manager {
    public int stockOptions;
}
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

Code:

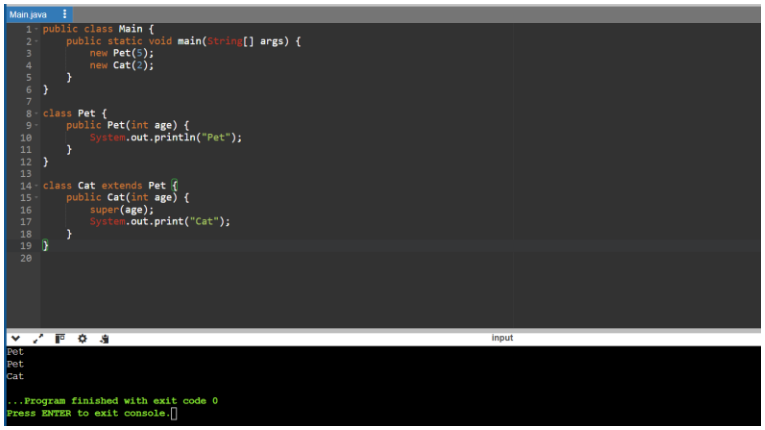
```
public class Main {
  public static void main(String[] args) {
    Employee employee = new Employee();
    Manager manager = new Manager();
    Director director = new Director();

    employee.salary = 50000;
    director.salary = 80000;
    manager.budget = 100000;
    director.stockOptions = 1000;
  }
}

class Employee {
  public int salary;
}

class Manager extends Employee {
  public int budget;
}

class Director extends Manager {
  public int stockOptions;
}
```

2. Will the code compile? If yes, determine the output. If not, state why.

```
class Pet {
    public Pet(int age) {
        System.out.print("Pet");
    }
}
public class Cat extends Pet {
    public Cat() {
        System.out.print("Cat");
    }
    public static void main(String[ ] args) {
        new Pet(5);
    }
}
```

Answer:

The constructor 'Pet' in the 'Cat' class cannot run because the actual and formal parameter lists differ in length. Also, the superclass constructor in the 'Cat' class should explicitly call its subclass constructor, 'Pet', to give the appropriate parameters. Additionally, I noted that the 'Cat' class has two opening curly braces, which suggests that there should be two closing curly braces, but only one is present. After troubleshooting the code, I explicitly called the parent class constructor and put a closing curly brace at the end of the 'Cat' class. These are the input and output results I acquired.



```
public class Main {
    public static void main(String[] args) {
        new Pet(5);
        new Cat(2);
    }
}

class Pet {
    public Pet(int age) {
        System.out.println("Pet");
    }
}

class Cat extends Pet {
    public Cat(int age) {
        super(age);
        System.out.print("Cat");
    }
}
```

```
Pet
Pet
Cat

...Program finished with exit code 0
Press ENTER to exit console.
```

Code:

```
public class Main {

  public static void main(String[] args) {
    new Pet(5);
    new Cat(2);
  }
}


class Pet {
  public Pet(int age) {
    System.out.println("Pet");
  }
}


class Cat extends Pet {
  public Cat(int age) {
    super(age);
    System.out.print("Cat");
  }
}
```