Test your subclasses by modifying the EmployeeTest class. Have your code do the following:

a. Remove the code that creates an instance of the "Jane Smith" Employee.

b. Create an instance of an Engineer with the following information:

Field	Choices or Values
ID	101
Name	Jane Smith
SSN	012-34-5678
Salary	120_345.27

c. Create an instance of a Manager with the following information:

Field	Choices or Values
ID	207
Name	Barbara Johnson
SSN	054-12-2367
Salary	109_501.36
Department	US Marketing

d.

Create an instance of an Admin with the following information:

Field	Choices or Values
ID	304
Name	Bill Munroe
SSN	108-23-6509
Salary	75_002.34

e. Create an instance of a Director:

Field	Choices or Values
ID	12
Name	Susan Wheeler
SSN	099-45-2340
Salary	120_567.36
Department	Global Marketing
Budget	1_000_000.00

f. Use the printEmployee method to print out information about each of your Employee objects.

g. (Optional) Use the raiseSalary and setName methods on some of your objects to make sure that those methods work.

h. Save the EmployeeTest class and test your work.

8. (Optional) Improve the look of the salary print output using the NumberFormat class.

a. In the printEmployee() method of Employee.java, use the following code to get an instance of a static java.text.NumberFormat class that you can use to format the salary to look like a standard US dollar currency:

```
NumberFormat.getCurrencyInstance().format((double)
getSalary()));
```

(Optional) Add additional business logic (data validation) to your `Employee` class.

a. Prevent a negative value for the `raiseSalary` method.

b. Prevent a null or empty value for the `setName` method.