

COP 3330

Programming Assignment 1: Receipt Generator



Program Overview

You will create a Java program that simulates a receipt generator for a fictional store. The program will:

- Generate a random receipt number and date.
- Accept user input for item names and prices.
- Calculate tax and total cost.
- Display a formatted receipt.

Functional Requirements

1. Receipt Number

Generate a random number between 1000 (inclusive) and 2000 (exclusive).

2. Date Generation

Randomly generate a date in the format:

Month Day Year

Constraints:

- Year: Between 1900 and 2099 (both of the years are included)
- Day: Between 1 and 28. Add 0 before digits[1-9] so that the field is always 2 digits. 28 is included. (1 and 28 is included)
- Month Name: The first letter will be a capital letter.

Tip: You are highly encouraged to write a separate function to generate the date. You may define the function prototype as you prefer.

3. User Input

Use the Scanner class to read item names and prices.

Continue accepting input until the user enters "DONE" as the item name. Only "DONE" with all uppercase letters will be perceived as stopping input.

4. Tax Rules

If the item name is exactly "Food" or "food", apply 0% tax.

For all other inputs, apply 30% tax.

Only the exact strings "Food" or "food" are recognized as tax-exempt food items. "hamburger", "aubergine", or any other food-related terms will be applied 30% tax.

5. Receipt Details

For each item, display:

- Item number
- Capitalized item name (capitalize the first letter if needed)
- Price including tax, formatted to two decimal places using:

```
double num = 11.234;
```

```
System.out.println("total " + String.format("%.2f", num));
```

Total price and item price should be formatted to two decimal places.

6. Total Calculation

Display the total number of items and the total cost, formatted to two decimal places.

7. Output Formatting

Follow the sample output format exactly.

Pay close attention to:

- Number of asterisks
- Spacing between labels and values (e.g., 6 spaces between "receipt number" and the number)
- Capitalization of item names

Submission Guidelines

- Your Java source file must begin with the following package declaration:

```
package assignment1;
```

- Do not use the default package.
- Include a main method in your source file.
- Submit only one **.java** file—do not submit your entire IDE project/class files.
- Do not use Java Collections or any third-party libraries.
- Your code must compile without errors. Non-compiling submissions will be penalized.
- Your program's output must exactly match the sample outputs, including formatting (e.g., number of stars, spaces, etc.).
- There are 6 spaces after *receipt number*. There are 6 spaces after *items* (before *total*). Besides there is only one space if there is space. No space between asterisks. *S Store* has space before and after asterisks.
- Do not concatenate more than one class in your java file. You will receive 0 if you do so.

Example Outputs. Inputs are written in bold.

Test Case 1

```
*****
***** S store *****
*****
receipt number    1406
November 10 2028
Write item name DONE
0 items    total 0.00
*****
*****
*****
```

Test Case 2

```
*****
***** S store *****
*****
receipt number    1363
February 14 1916
Write item name Chair
Write price 14.0004
item number 1 Chair 18.20
Write item name DONE
1 items    total 18.20
*****
*****
*****
```

Test Case 3

```
*****
***** S store *****
*****
receipt number    1113
November 26 1951
Write item name food
Write price 145
item number 1 Food 145.00
Write item name shoe
Write price 142.2578
item number 2 Shoe 184.94
Write item name T-shirt
Write price 0.45
item number 3 T-shirt 0.59
Write item name DONE
3 items    total 330.52
*****
*****
*****
```

Evaluation Criteria

Your submission will be assessed based on the following criteria:

- Test case 1: 20 pts
- Test case 2: 20 pts
- Test case 3: 20 pts
- Receipt Number Generation: 10 pts
 - Random number between 1000 (inclusive) and 2000 (exclusive)
- Date Generation: 10 pts
 - Random date in format Month Day Year with:
 - Year: 1900 – 2099
 - Day: 01 – 28
 - Month: Full name of the month, first letter capitalized.
 - Implemented via a separate function
- Code Quality: 10 pts
 - Code is well-organized, readable, and includes meaningful comments. If there is no header comment, you will lose 10 pts. Header comment should include your name lastname,PID, assignment number and in a few sentences what the code does.
- Package name: 10 pts
 - If the package name is stated as assignment1
- Non compiling submissions will receive 0 from test cases.

IF YOU SEND CLASS FILES OR IF YOUR FILE IS NOT READABLE, YOU WILL RECEIVE 0. NO EXCEPTIONS FOR THIS RULE.

Randomly generated numbers/lines will not be used in test case grading.

Total: 100 points