# Tutorial of ArrayList and String

> Based on the tutorial of "2020S-Java-A" designed by teaching group in SUSTech
>
> Modified (only change to markdown file) by ZHU Yueming in 2021. April. 6th
>
> Update by ZHU Yueming in 2021. Nov. 1st. Add File read and write demo
>
> Modified(Part of the content change from word to markdown ) by JIA Yanhong in 2022. Oct. 27th

## Objective

- Learn to use ArrayList
- Learn to use various String methods

## Part 1: ArrayList

### Before Exercise

**Step 1: How to manage multiple circle objects ?**

We can use an array or an `ArrayList` to manage them.

In the main method, create an arrayList with a Circle type, to store many objects of Circle. Add the following code at the end of main method.

```
ArrayList<Circle> circleList=new ArrayList<Circle>();
circleList.add(c1);
System.out.printf("Radius of %d circle is %.2f: \n", 1 ,
circleList.get(0).getRadius());
```

Sample output:

```
5.0
The area of c1 is 78.54
The perimeter of c1 is 31.42
Position of the circle is (0.0,0.0)
Radius of 1
circle is 5.00:
```

**Step 2: Add more circles in the ArrayList**

Add the following code at the end of main method.

```
for(int i=1;i<5;i++) {
      circleList.add(new Circle());
      circleList.get(i).setRadius(i);
      circleList.get(i).setX(Math.random()*5);
      circleList.get(i).setY(Math.random()*5);
    }

    System.out.println("---Begin to print the circle list---");
    for(int i=0;i<5;i++) {
      System.out.printf("The area of %d circle is %.2f\n",
          i+1, circleList.get(i).area());
      System.out.printf("The perimeter is %.2f\n",
          circleList.get(i).perimeter());
    }
```

Sample output:

```
5.0 The area of c1 is 78.54
The perimeter of c1 is 31.42
Position of the circle is (0.0,0.0)
Radius of 1 circle is 5.00:
---Begin to print the circle list--
The area of 1 circle is 78.54
The perimeter is 31.42
The area of 2 circle is 3.14
The perimeter is 6.28
The area of 3 circle is 12.57
The perimeter is 12.57
The area of 4 circle is 28.27
The perimeter is 18.85
The area of 5 circle is 50.27
The perimeter is 25.1
```

## Exercise 1 : Food

Design a class named **Food**. The class contains:

- Private data fields:

  int **id**;

  String **name;**

  String **type**;

  int **size**;

  double **price;**
- Implement a public method named `getMenu()` to print all the information of this food object.
- Implement the **getter** and **setter** method for each private field of Food.

In `FoodTest` class, create four objects of Food as follows:

| Object Name | id | name | type | size | price |
|---|---|---|---|---|---|
| **pizza1** | 1 | pizza | Seafood | 11 | 12 |
| **pizza2** | 2 | pizza | Beef | 9 | 10 |
| **Fried rice** | 3 | fried rice | Seafood | 5 | 12 |
| **Noodles** | 4 | noodles | Beef | 6 | 14 |

Create an `ArrayList<Food>` to add those four Food objects, and then show the information of them as follows by iterating the `ArrayList<Food>` we created.

```
Seafood pizza: (11 Inches) 120.00 $
Beef pizza: (9 Inches) 100.00 $
Seafood fried rice: (5 Inches) 40.00 $
Beef noodle: (6 Inches) 35.00 $
```

## Exercise 2: Combine Food and User

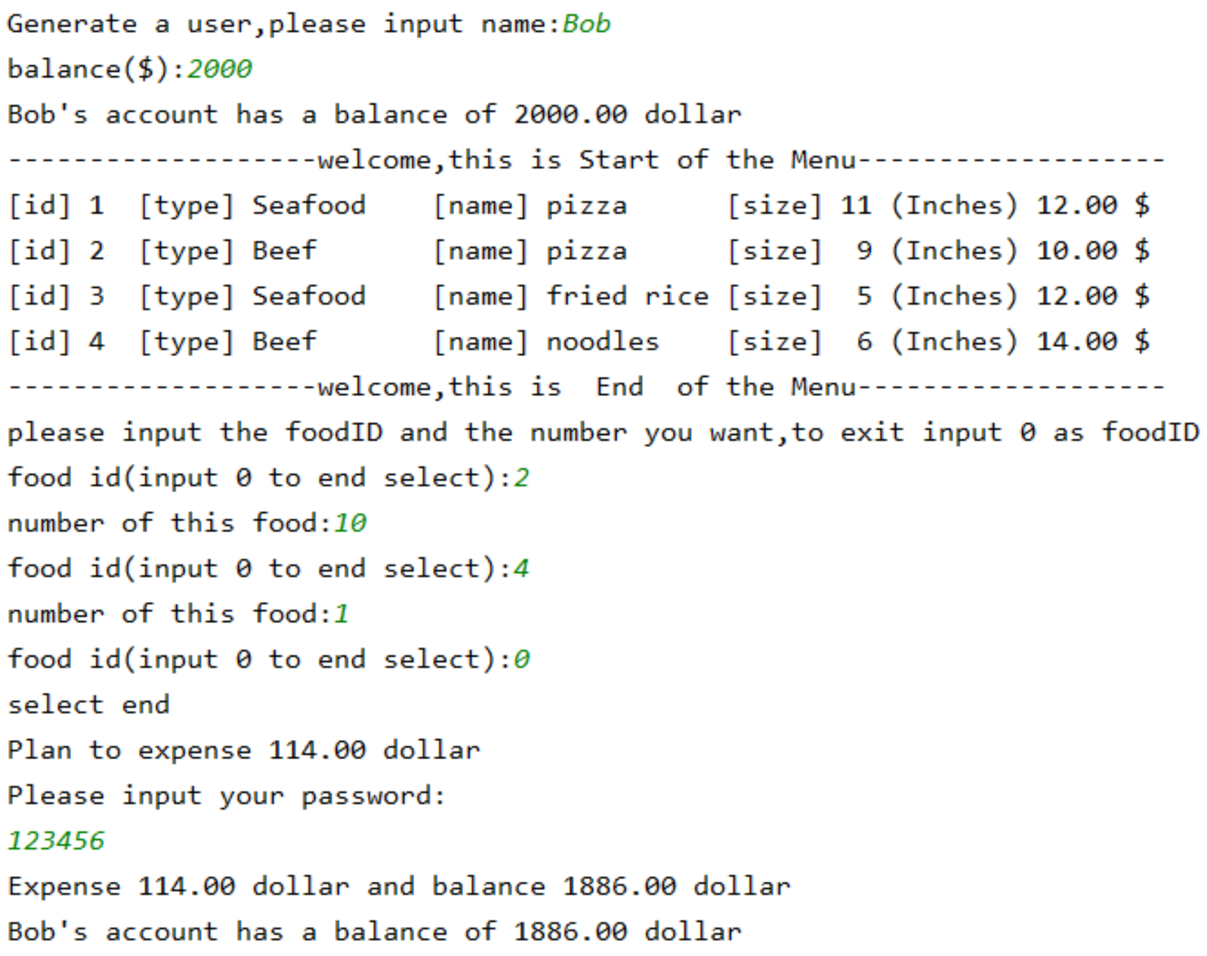
Design a class named `SoftOpening`. The class contains no data fields but:

- Implement a public static method named `generateMenu()` to generate 4 object of Food and add them to the `ArrayList<Food>`.
- Implement a public static method named `getMenu(ArrayList<Food> foodList)` to print the items in the `ArrayList<Food>` as designed.
- Implement a public static method named to `generateUser(Scanner in)` to generated a user whose account and money is get by using the Scanner object 'in'.
- Implement a public static method named `UserConsume(ArrayList<Food> foodList, User user, Scanner in)` to invoke the `getMenu()`, ask user to select the foods in the Menu, count the cost and invoke the expense of the user.
- Invoke the method `introduce()` of the User object to show his/hers balance.

**Statements in main method**:

```
Scanner in = new Scanner(System.in);
ArrayList<Food> foodList = genarateMenu();  //generate a Menu
User user = genarateUser(in);              //generate a user
user.introduce();                          //show the account of the user
userConsume(foodList,user,in);             //user consume
user.introduce();                          //show the account of the user
in.close();
```

Sample Output:



```
Generate a user,please input name:Bob
balance($):2000
Bob's account has a balance of 2000.00 dollar
-------------------welcome,this is Start of the Menu------------------
[id] 1  [type] Seafood    [name] pizza      [size] 11 (Inches) 12.00 $
[id] 2  [type] Beef       [name] pizza      [size]  9 (Inches) 10.00 $
[id] 3  [type] Seafood    [name] fried rice [size]  5 (Inches) 12.00 $
[id] 4  [type] Beef       [name] noodles    [size]  6 (Inches) 14.00 $
-------------------welcome,this is  End  of the Menu------------------
please input the foodID and the number you want,to exit input 0 as foodID
food id(input 0 to end select):2
number of this food:10
food id(input 0 to end select):4
number of this food:1
food id(input 0 to end select):0
select end
Plan to expense 114.00 dollar
Please input your password:
123456
Expense 114.00 dollar and balance 1886.00 dollar
Bob's account has a balance of 1886.00 dollar
```

# Part 2: String manipulations

Please use String methods

(https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/String.html) to finish the tasks below.

Methods in the Character class are also helpful:

https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/Character.html.

## Exercise 3

Write a program to check if a string provided by a user is a palindrome or not. A string is a palindrome is the reverse of the string is the same as the string (we do not differentiate upper-case and lower-case letters in this task). For example, “abba” is a palindrome. “#Aa#” and “0” are also palindromes.

Your program should continuously take user inputs for checking and stops when the user types “quit”.

A sample run:

```
Type a string ("quit" to finish): hello
hello is not a palindrome
Type a string ("quit" to finish): many
many is not a palindrome
Type a string ("quit" to finish): 0
0 is a palindrome
Type a string ("quit" to finish): 900
900 is not a palindrome
Type a string ("quit" to finish): #Aa#
#Aa# is a palindrome
Type a string ("quit" to finish): quit

Process finished with exit code 0
```

## Exercise 4

Write a program to remove all repeated characters in a string provided by the user and return a new string without any repeating characters or white spaces. Please use `StringBuilder` to build the new string.

Sample runs:

```
Please type a string: hello
After removing repeating chars and spaces: helo
```

```
Please type a string:
Empty string, exit…
```

```
Please type a string: abcd  bcde cdef
After removing repeating chars and spaces: abcdef
```

## Exercise 5

Write a program to count the occurrence of a substring in a string. The program should ask the user to input two strings s1 and s2, and output the number of occurrences of s2 in s1.

Sample runs:

```
s1: JavaExamplesJavaCodeJavaProgram
s2: Java
Found at index: 0
Found at index: 12
Found at index: 20
Total occurrences: 3

s1: abcd  bcde cdef
s2: bc
```

```
Found at index: 1
Found at index: 6
Total occurrences: 2

s1: abcdefg
s2: xyz
Total occurrences: 0
```

# API References:

## String methods:

[https://docs.oracle.com/javase/8/docs/api/java/lang/String.html](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html)

```
 public int length()

 public char charAt(int index)

 public boolean startsWith(String prefix)

 public boolean equals(Object anObject)

 public boolean equalsIgnoreCase(String anotherString)

 public String trim()

 public int indexOf(String str)

 public int indexOf(String str, int fromIndex)

 public String substring(int beginIndex)

 public String substring(int beginIndex, int endIndex)

 public String[] split(String regex)

 public char[] toCharArray()
```

## Character methods:

[https://docs.oracle.com/javase/8/docs/api/java/lang/Character.html](https://docs.oracle.com/javase/8/docs/api/java/lang/Character.html)

```
public static char toLowerCase(char ch)

public static boolean isWhitespace(char ch)
```

## StringBuilder methods:

https://docs.oracle.com/javase/8/docs/api/java/lang/StringBuilder.html

```
public StringBuilder append(char c)


public String toString()
```