* Homework #3

**# Due date: 16th/Oct, 23:59:59**

**# Put your answer in “Your Code)” section in each question**

**# For each solution you wrote, you MUST give comments for important lines. Without comments, your solutions will be considered copy&paste codes.**

**# You also need to provide a screenshot of your result (i.e., Logcat output).**

**# You must use Kotlin language.**

**1. Write a code to check the number of each alphabe in the given string.**

**- Use map**

**Code base)**

val sequence = "abcabcdefabc"

// implement here

**Output example)**

텍스트이(가) 표시된 사진

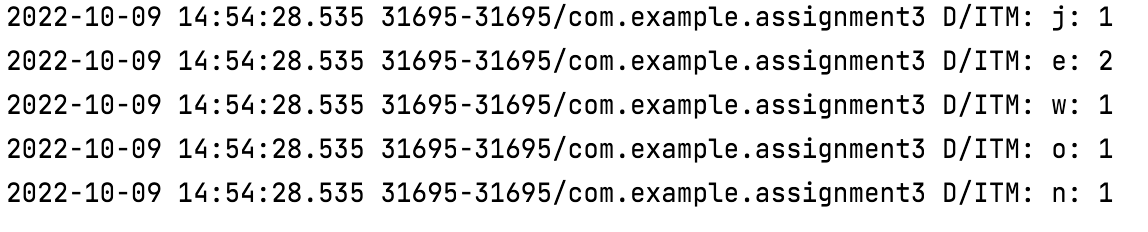
자동 생성된 설명

**Your Code)**

*// Initializing a mutable map* **val** checkNum = *mutableMapOf*<String, Int>()  
  
*// Starting to check from the first letter to the last letter* **for**(i **in** 0..(sequence.**length**-1)){  
*// Checking repetition, if there isn't a repetition, add the key and value in the mutable map* **if**(!checkNum.*contains*(**"sequence[i]"**)){  
*// Counting the specific letter in the word(sequence)* **val** temp = sequence.*count***{  
 it** == sequence[i]  
 **}***// Putting the key and value* checkNum.put(sequence[i].toString(), temp)  
 }  
 }  
  
*// Getting a result of key and value using iteration* **for**((k, v) **in** checkNum){  
 Log.d(**"ITM"**, **"$**k**: $**v**"**)  
 }

**Your Output)**

**val** sequence = **"jeewon"**

****

**2. Write a Data Class (e.g., Item class) to represent the following information:**

* **Name property (String type)**
* **Price property (Int type)**
  + **When this property is set, a message like “price set to [xxxx]. Are you serious?” must be printed out.**
* **Share property (Int type)**
* **Whenver a new instance is created, a message like “[name] item was created.” Must be printed out**

**Test code)**

val item1 = Item(name="jinwoo1").*apply***{** share = 100  
 price = 500  
**}**

**Output example)**

텍스트, 오렌지, 어두운이(가) 표시된 사진

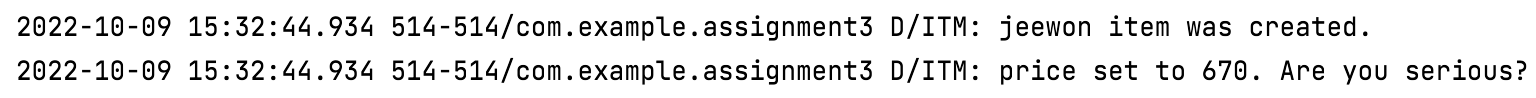
자동 생성된 설명

**Your Code)**

*//Initializing Item class***data class** Item(**val name**: String) {  
*// As soon as this class happens, init begins* **init** {  
 Log.d(**"ITM"**, **"$name item was created."**)  
 }  
  
*// Initializing share* **var share**:Int = 0  
  
 *// Initializing price* **var price**: Int = 0  
 **set**(value){  
 **field** = value  
 *// When we set the value of price, this message will be printed* Log.d(**"ITM"**, **"price set to $**value**. Are you serious?"**)  
 }  
  
}

**Your Output)**

**val** item1 = Item(name=**"jeewon"**).*apply***{  
 share** = 240  
 **price** = 670  
**}**

****

**3. Create a List to include 10 randomly generated Item instances (the result of Question #2).**

**- Use for loop to create and add items into your list**

**- The name field should be “your name”+iteration number (1..10)**

**- The share filed should be 100**

**- the price field should be randomly set (0~1000)**

**Then, use forEach() method to print items in your list.**

**Output example)**

테이블이(가) 표시된 사진

자동 생성된 설명

**Your Code)**

*// Initializing a list* **var** randomList = *mutableListOf*<Item>()  
  
*// Creating and adding items into list(randomList)* **for**(i **in** 1..10){  
*// Setting my name as jeewon plus iteration number* randomList.add(Item(name = **"jeewon$**i**"**).*apply***{***// Setting share as 100* **share** = 100  
*// Setting price as random integer between 0 and 1000* **price** = Random().nextInt(1000)  
 **}**)  
 }  
   
*// Printing items in my list* randomList.*forEach***{** Log.d(**"ITM"**,**"name: ${it**.**name} price: ${it**.**price}"**)  
 **}**

**Your Output)**

**Text

Description automatically generated**

**4. Then, find the instances whose price is higher than 500 from your list.**

**- Use filter() method**

**- No for loop is allowed**

**Output example)**



**Your Code)**

**val** filteredList = randomList.*filter***{***// applying filter* **it**.**price**>500  
 **}***// Showing result* Log.d(**"ITM"**,**"${**filteredList**}"**)

**Your Output)**



**4. Finally, sort your list based on the price and print the sorted list using 1) run statement and 2) also statement.**

**- In run statement, you need to add code to sort the list based on the price, and get the string represetation of the list**

**- In also statement, you need to printout the list with uppercase letters only**

**Base code)**

val str = myList.*run* **{**

// sort the list

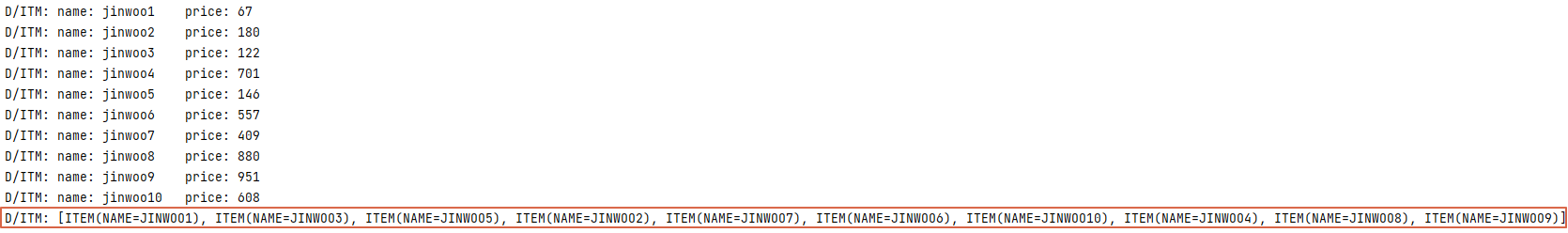
// return the string representation of the list

**}**.*also* **{**

// print the list contents with capital letters

**}**

**Output example)**



**Your Code)**

**val** str = randomList.*run***{***// Sorting  
 sortedBy***{it**.**price}**.toString()  
 **}**.*also***{***// Printing uppercase result* Log.d(**"ITM"**,**"${it**.*uppercase*()**}"**)  
 **}**

**Your Output)**

Text

Description automatically generated with medium confidence