Gebze Technical University
Computer Engineering

CSE 222 - 2019 Spring

HOMEWORK 2 REPORT

Hamza YOĞURTCUOĞLU
171044086

Course Assistant: Arş. Gör. Ayşe ŞERBETÇİ TURAN

1 INTRODUCTION

1.1 Problem Definition

I have encountered three basic problems without implementing this project.

I will give you information on how to solve.

● Design a good design to make the most efficient implementation.

● Where to Experiments in the ExperimentList and linking each other to the system. (NEXT,NEXTDAY)

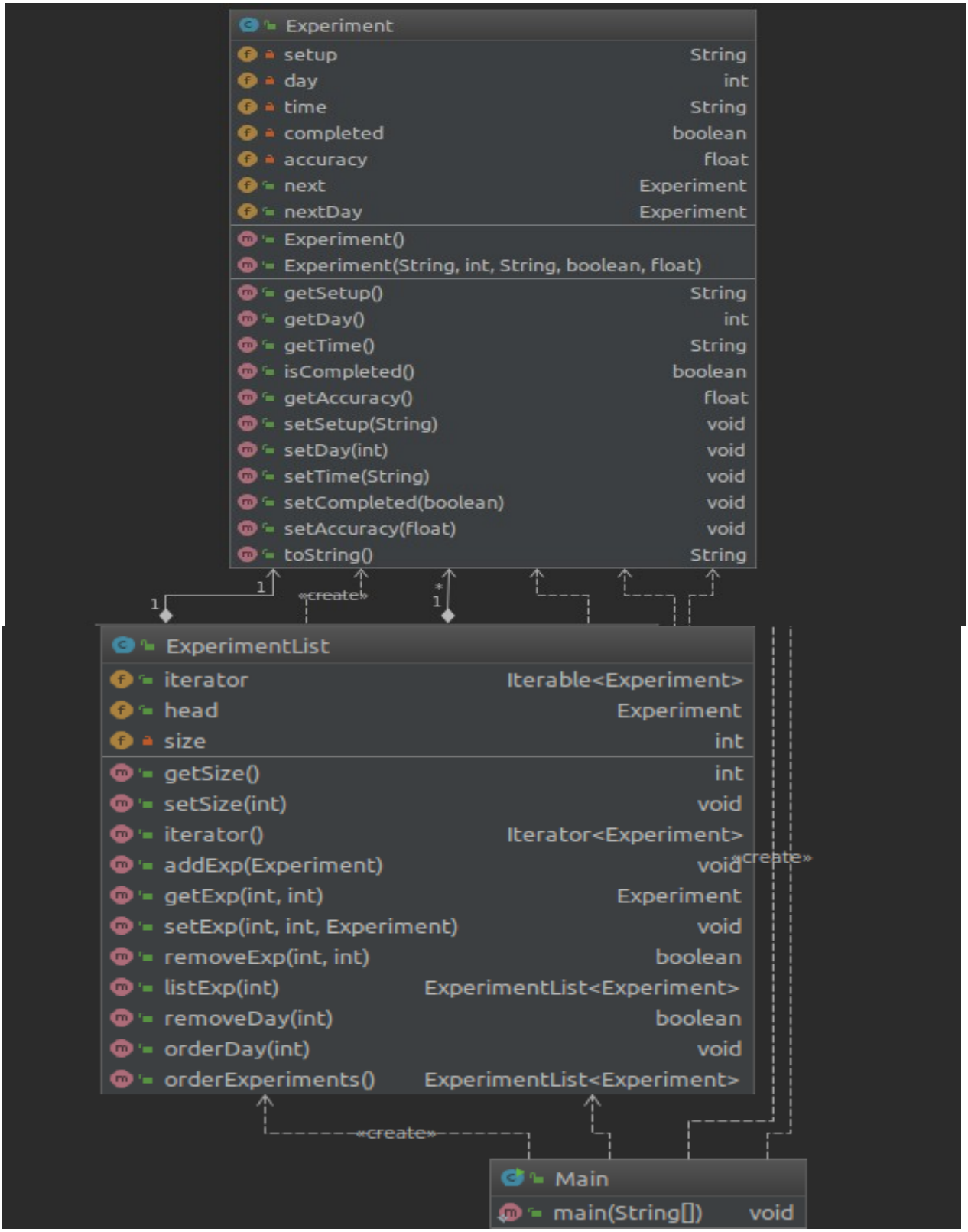
● Remove the data structure(Linkedlist) to be used to remove Experiment the in the Linkledlist.

1.2 System Requirements

- System must have two class whose are ExperimentList and Experiment Class.
- Experiment class has to be itreable that means . It can be move next in the ExperimentList
- ExperimentList (LinkList Class) has to be below properties :
    + addExp(Experiment): insert experiment to the end of the day
    + getExp(day, index) : get the experiment with the given day and position
    + setExp(day, index,) set the experiment with the given day and position
    + removeExp(day, index): remove the experiment specified as index from given day
    + listExp(day): list all completed experiments in a given day
    + removeDay(day): remove all experiments in a given day
    + orderDay(day): sorts the experiments in a given day according to the accuracy, the changes will be done on the list
    + orderExperiments(): sorts all the experiments in the list according to the accuracy, the original list should not be changed since the day list may be damage
- All day , index would be checked if given day or index is out of bound , it must give error message.
- Experiment Class has to be below properties :
    +setup (String): explains the experimental setup
    +day(integer): represents the day of start
    +time(Time): represents the time of start
    +completed(boolean): indicates whether it is completed or not
    +accuracy(float): represents the output (not a valid value if the experiment is not completed)

2 METHOD

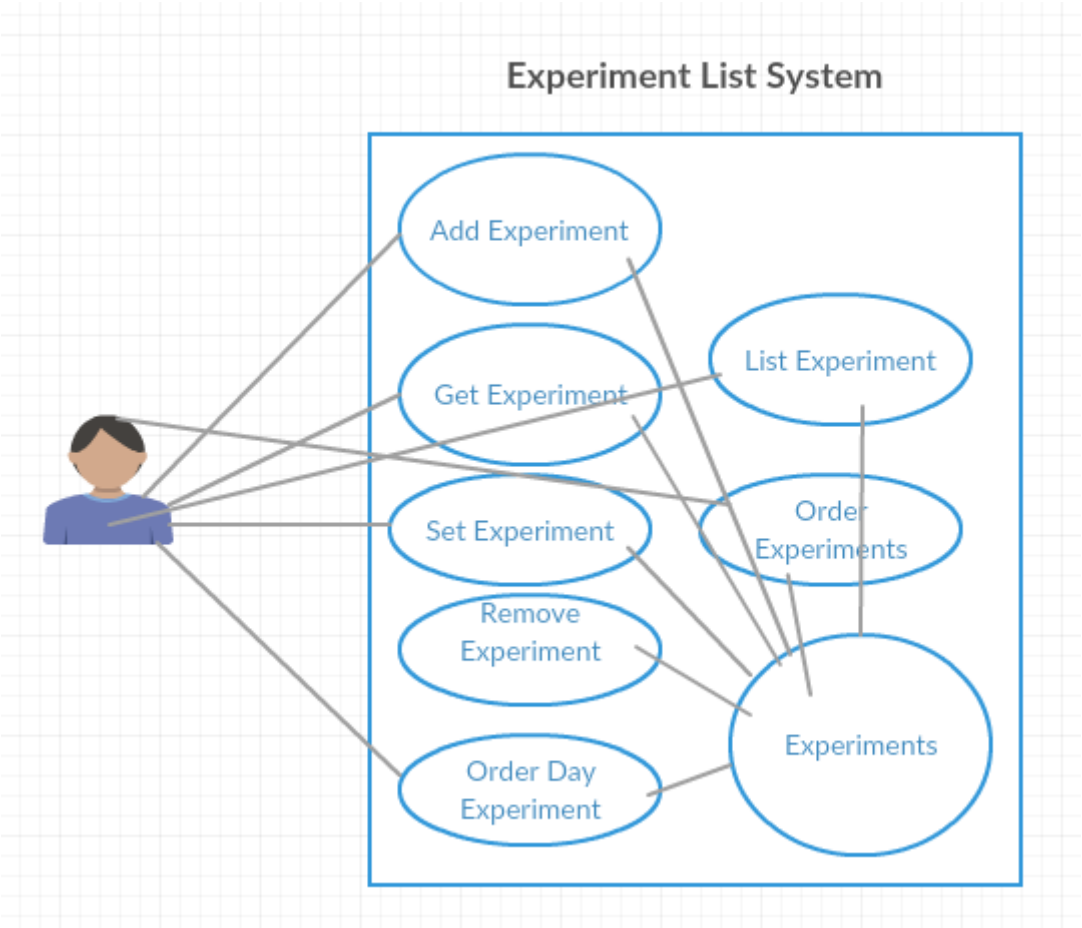
2.1 Class Diagrams



1) All ExperimentList and Experiments are created in main class.

2) Just a creating LinkList that can be have many Experiments Objects.

2.2 Use Case Diagrams



Users can realize above Actions through Experiment.

System can do Actions those are Add Experiment, Get Experiment , Set Experiment, Remove Experiment,Order Day Experiment , List Experiment , Order Experiment through Experiments.

2.3 Problem Solution Approach

In order to solve the problem, I have carried out the following operations.

+How to store in Link List with hierarcy between Linklist and NODE.

+I'll use my own Link List to keep the Experiments. Using this data structure will allow me to easily access the data in the list. It will also provide convenience.

+Order day and Order Experiments are sorted by accuracy that is sorted through Experiments Array.

+ How to Next DAY and next reference each will be checked when new experiment is added.

+When All remove or listed operation will be used Temp Experiment that is previous Experiment is retain for needs.

3 RESULT

----------Complexity For All Functions----------

Functions	Complexity
addExp	O(n)
getExt	log(n)
setExp	log(n)
removeExp	O(n)
listExp	O(n)
removeDay	O(n)
orderDay	n.log(n)
orderExperiments	n.log(n)

Add Function Complexity O(n)

```
for (int i = 0; i < size && pre != null && (pre.getDay() != exp.getDay()); i++) {

    if (exp.getDay() > pre.getDay()) {
        if (pre.next != null) {
            if (exp.getDay() > pre.next.getDay())
                pre = pre.next;
        }
    }
}
```

When each adding experiment. Each experiments are passed then added just a for loop .

Get Function Complexity O(logn)

```
while (flag == 0 && NEXT.nextDay != null) {

    if (NEXT.nextDay.getDay() == day) {
        flag = 1;
        NEXT = NEXT.nextDay;
    } else {
        NEXT = NEXT.nextDay;
    }
}
for (int i = 1; i < index + 1; i++) {
    if (NEXT.next.getDay() == day)
        NEXT = NEXT.next;
    else
        throw new IndexOutOfBoundsException();
```

Through Nextday refenrence Head of need day than forwarding to needed index.

Set Function Complexity O(logn) : That is like a same concept with get function

Remove Function Complexity O(n) : Each situation controled head ,tail or any day .

N experiments are iterated for removing experiment. IF not head or tail . Theta complexity is -------->log n <--------

List Experiment Function Complexity O(n)

Each head of days a NEXTDAY reference for forwarding day. Then Days of experiments (n)

Remove Day Function Complexity O(n)

forwarding through nextday refecence . Previous of Next of Head of Day and Next Day are keeping for seperate linklist. It will be log(n) complexity but first head of day will keep n complexity then worst complexity (n)

Orderday Day Function Complexity n.logn
MergeSort is used.(nlog)

orderExperiments fuction Complexity n.logn
MergeSort is used.

3.1 Test Cases

ADDEXP : ADDING EXPERIMENT ACCRODING TO DAY

```
System.out.println("\n------>ADD EXPERIMENTS<------");
ExpList.addExp(new Experiment( setup: "setup1-a", day: 1,time, completed: true, accuracy: 55.55f));
ExpList.addExp(new Experiment( setup: "setup4-a", day: 4,time, completed: false, accuracy: 92.55f));
ExpList.addExp(new Experiment( setup: "setup3-a", day: 3,time, completed: true, accuracy: 78.56f));
ExpList.addExp(new Experiment( setup: "setup4-b", day: 4,time, completed: true, accuracy: 95.55f));
ExpList.addExp(new Experiment( setup: "setup5-a", day: 5,time, completed: false, accuracy: 20.55f));
ExpList.addExp(new Experiment( setup: "setup2-a", day: 2,time, completed: false, accuracy: 58.55f));
ExpList.addExp(new Experiment( setup: "setup2-b", day: 2,time, completed: true, accuracy: 78.55f));
ExpList.addExp(new Experiment( setup: "setup1-b", day: 1,time, completed: true, accuracy: 13.28f));
ExpList.addExp(new Experiment( setup: "setup4-c", day: 4,time, completed: true, accuracy: 94.55f));
ExpList.addExp(new Experiment( setup: "setup3-b", day: 3,time, completed: false, accuracy: 90.55f));
ExpList.addExp(new Experiment( setup: "setup4-d", day: 4,time, completed: true, accuracy: 91.55f));
ExpList.addExp(new Experiment( setup: "setup5-b", day: 5,time, completed: true, accuracy: 49.14f));
ExpList.addExp(new Experiment( setup: "setup2-c", day: 2,time, completed: false, accuracy: 48.55f));
ExpList.addExp(new Experiment( setup: "setup2-d", day: 2,time, completed: true, accuracy: 10.55f));
```

GETEXP : GETTING DAY 1 AND INDEX IS 1

```
System.out.println("\n------>GET EXPERIMENT<------");
System.out.println(ExpList.getExp( day: 1, index: 1));
System.out.println("------>GET EXPERIMENT<------\n");
```

SETEXP :SETTING DAY 4 AND INDEX 1

```
System.out.println("\n------>SET EXPERIMENT<------");
Experiment setEX=new Experiment( setup: "setup4-k", day: 4,time, completed: true, accuracy: 1.55f);
System.out.println(setEX);
ExpList.setExp( day: 4, index: 1,setEX);
System.out.println("------>SET EXPERIMENT<------\n");
```

REMOVEEXP : REMOVING 5 DAY INDEX 0

```
System.out.println("\n------>REMOVED EXPERIMENT<------");
System.out.println(ExpList.removeExp( day: 5, index: 0));
Iterator iter2 = ExpList.iterator();
while (iter2.hasNext()){
    System.out.println(iter2.next());
}
System.out.println("------>REMOVED EXPERIMENT<------\n");
```

LISTEXP : RETURN LISTED EXPERIMENTS DAY : 2

```
System.out.println("\n------>LIST EXPERIMENTS<------");
ExperimentList<Experiment> dayCompleted= ExpList.listExp(2);
Iterator iter9 = dayCompleted.iterator();
while (iter9.hasNext()){
    System.out.println(iter9.next());
}
System.out.println("\n------>LIST EXPERIMENTS<------");
```

REMOVEDAY : REMOVE DAY IN LINKLIST DAY :1

```
System.out.println("\n------>REMOVEDAY<------");
ExpList.removeDay(1);
Iterator iter100 = ExpList.iterator();
while (iter100.hasNext()){
    System.out.println(iter100.next());
}
System.out.println("\n------>REMOVEDAY<------");
```

ORDERDAY: DAY 4

```
System.out.println("\n------>ORDER DAY<------");
ExpList.orderDay(4);
Iterator iter10 = ExpList.iterator();
while (iter10.hasNext()){
    System.out.println(iter10.next());

}
System.out.println("\n------>ORDER DAY<------");
```

ORDEREXPERIMENT : AL EXPERIMENTS ARE SORTED

```
System.out.println("\n------>ORDER EXPERIMENTS<------");
ExperimentList orderExperiment=ExpList.orderExperiments();
Iterator iter101 = orderExperiment.iterator();
while (iter101.hasNext()){
    System.out.println(iter101.next());
}
System.out.println("\n------>ORDER EXPERIMENTS<------");
```

3.2 Running Results

ALL RESULT IS CHECKED ACCORDING TO ABOVE TESTS

```
------>ADD EXPERIMENTS<------
Experiment{setup='setup1-a', day=1, time='04:36:39', completed=true, accuracy=55.55}
Experiment{setup='setup1-b', day=1, time='04:36:39', completed=true, accuracy=13.28}
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup2-b', day=2, time='04:36:39', completed=false, accuracy=48.55}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}
Experiment{setup='setup2-d', day=2, time='04:36:39', completed=false, accuracy=58.55}
Experiment{setup='setup3-a', day=3, time='04:36:39', completed=false, accuracy=90.55}
Experiment{setup='setup3-b', day=3, time='04:36:39', completed=true, accuracy=78.56}
Experiment{setup='setup4-a', day=4, time='04:36:39', completed=true, accuracy=91.55}
Experiment{setup='setup4-b', day=4, time='04:36:39', completed=true, accuracy=94.55}
Experiment{setup='setup4-c', day=4, time='04:36:39', completed=true, accuracy=95.55}
Experiment{setup='setup4-d', day=4, time='04:36:39', completed=false, accuracy=92.55}
Experiment{setup='setup5-a', day=5, time='04:36:39', completed=true, accuracy=49.14}
Experiment{setup='setup5-b', day=5, time='04:36:39', completed=false, accuracy=20.55}
------>ADD EXPERIMENTS<------


------>REMOVED EXPERIMENT<------
true
Experiment{setup='setup1-a', day=1, time='04:36:39', completed=true, accuracy=55.55}
Experiment{setup='setup1-b', day=1, time='04:36:39', completed=true, accuracy=13.28}
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup2-b', day=2, time='04:36:39', completed=false, accuracy=48.55}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}
Experiment{setup='setup2-d', day=2, time='04:36:39', completed=false, accuracy=58.55}
Experiment{setup='setup3-a', day=3, time='04:36:39', completed=false, accuracy=90.55}
Experiment{setup='setup3-b', day=3, time='04:36:39', completed=true, accuracy=78.56}
Experiment{setup='setup4-a', day=4, time='04:36:39', completed=true, accuracy=91.55}
Experiment{setup='setup4-b', day=4, time='04:36:39', completed=true, accuracy=94.55}
Experiment{setup='setup4-c', day=4, time='04:36:39', completed=true, accuracy=95.55}
Experiment{setup='setup4-d', day=4, time='04:36:39', completed=false, accuracy=92.55}
Experiment{setup='setup5-b', day=5, time='04:36:39', completed=false, accuracy=20.55}
------>REMOVED EXPERIMENT<------
```

```
------>GET EXPERIMENT<------
Experiment{setup='setup1-b', day=1, time='04:36:39', completed=true, accuracy=13.28}
------>GET EXPERIMENT<------


------>SET EXPERIMENT<------
Experiment{setup='setup4-k', day=4, time='04:36:39', completed=true, accuracy=1.55}
------>SET EXPERIMENT<------
```

```
------>LIST EXPERIMENTS<------
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}

------>LIST EXPERIMENTS<------

------>REMOVEDAY<------
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup2-b', day=2, time='04:36:39', completed=false, accuracy=48.55}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}
Experiment{setup='setup2-d', day=2, time='04:36:39', completed=false, accuracy=58.55}
Experiment{setup='setup3-a', day=3, time='04:36:39', completed=false, accuracy=90.55}
Experiment{setup='setup3-b', day=3, time='04:36:39', completed=true, accuracy=78.56}
Experiment{setup='setup4-a', day=4, time='04:36:39', completed=true, accuracy=91.55}
Experiment{setup='setup4-k', day=4, time='04:36:39', completed=true, accuracy=1.55}
Experiment{setup='setup4-c', day=4, time='04:36:39', completed=true, accuracy=95.55}
Experiment{setup='setup4-d', day=4, time='04:36:39', completed=false, accuracy=92.55}
Experiment{setup='setup5-b', day=5, time='04:36:39', completed=false, accuracy=20.55}

------>REMOVEDAY<------
```

```
------>ORDER DAY<------
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup2-b', day=2, time='04:36:39', completed=false, accuracy=48.55}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}
Experiment{setup='setup2-d', day=2, time='04:36:39', completed=false, accuracy=58.55}
Experiment{setup='setup3-a', day=3, time='04:36:39', completed=false, accuracy=90.55}
Experiment{setup='setup3-b', day=3, time='04:36:39', completed=true, accuracy=78.56}
Experiment{setup='setup4-c', day=4, time='04:36:39', completed=true, accuracy=95.55}
Experiment{setup='setup4-d', day=4, time='04:36:39', completed=false, accuracy=92.55}
Experiment{setup='setup4-a', day=4, time='04:36:39', completed=true, accuracy=91.55}
Experiment{setup='setup4-k', day=4, time='04:36:39', completed=true, accuracy=1.55}
Experiment{setup='setup5-b', day=5, time='04:36:39', completed=false, accuracy=20.55}

------>ORDER DAY<------

------>ORDER EXPERIMENTS<------
Experiment{setup='setup4-c', day=4, time='04:36:39', completed=true, accuracy=95.55}
Experiment{setup='setup4-d', day=4, time='04:36:39', completed=false, accuracy=92.55}
Experiment{setup='setup4-a', day=4, time='04:36:39', completed=true, accuracy=91.55}
Experiment{setup='setup3-a', day=3, time='04:36:39', completed=false, accuracy=90.55}
Experiment{setup='setup3-b', day=3, time='04:36:39', completed=true, accuracy=78.56}
Experiment{setup='setup2-c', day=2, time='04:36:39', completed=true, accuracy=78.55}
Experiment{setup='setup2-d', day=2, time='04:36:39', completed=false, accuracy=58.55}
Experiment{setup='setup2-b', day=2, time='04:36:39', completed=false, accuracy=48.55}
Experiment{setup='setup5-b', day=5, time='04:36:39', completed=false, accuracy=20.55}
Experiment{setup='setup2-a', day=2, time='04:36:39', completed=true, accuracy=10.55}
Experiment{setup='setup4-k', day=4, time='04:36:39', completed=true, accuracy=1.55}

------>ORDER EXPERIMENTS<------
```

- Main titles -> 16pt , 2 line break
- Subtitles -> 14pt, 1.5 line break
- Paragraph -> 12pt, 1.5 line break