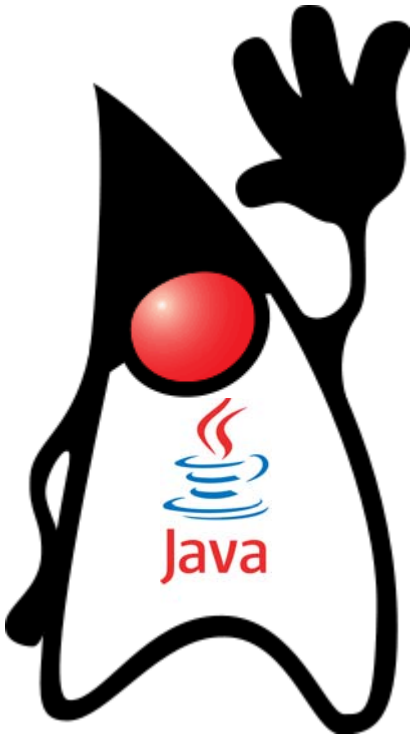
Arrays

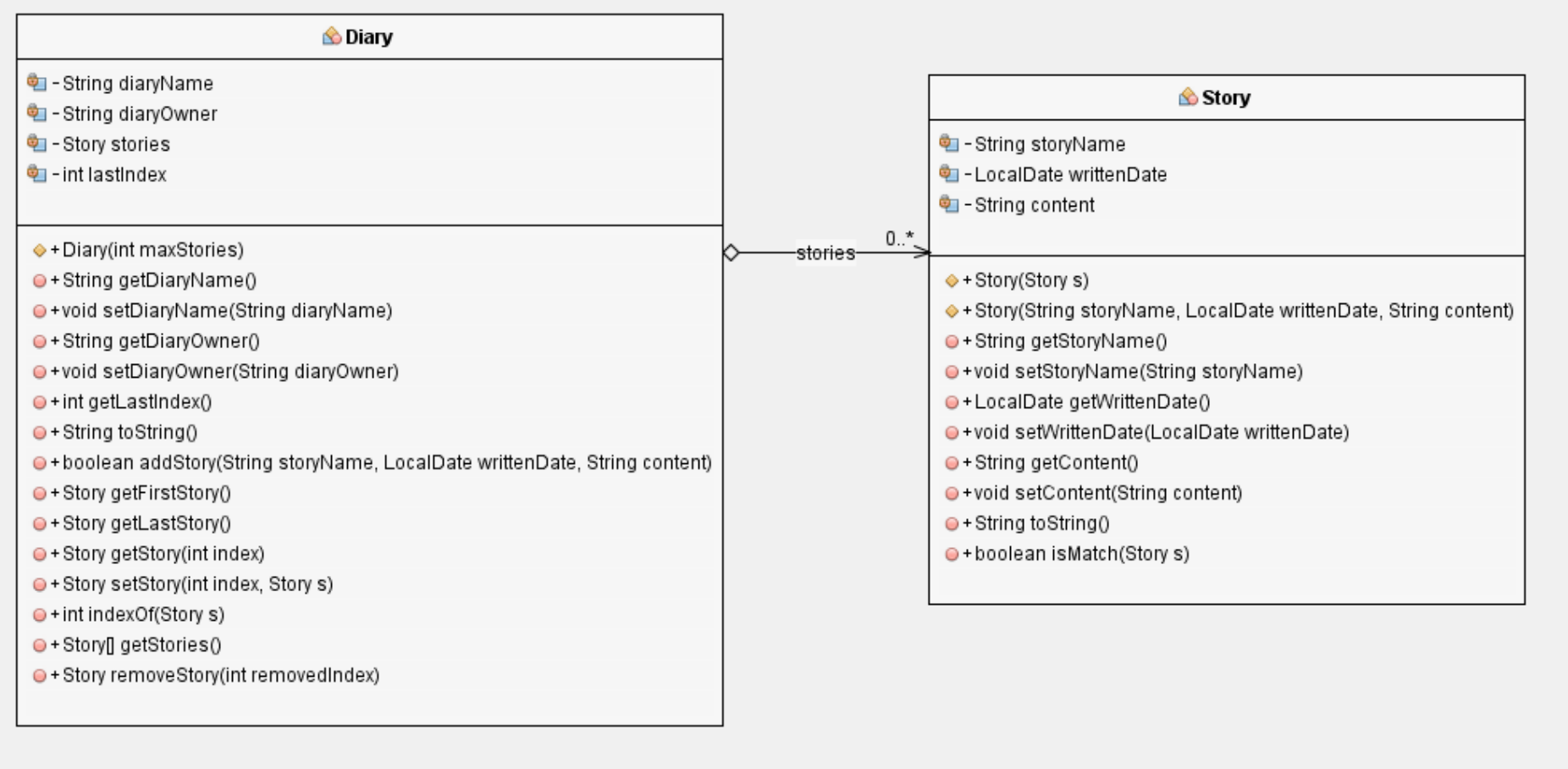


Asst.Prof.Dr. Umaporn Supasitthimethee

DiaryStory Requirements

- A diary contains stories.
  - Each story in the diary should have a name of the story, a date that the story is written, and the content of the story
  - You should be able to set and get the name, the date, and the content of the story
- A diary should have a name and an owner.
- You should be able to set and get the name and the owner of the diary.
- You can add a story into the diary and remove story from the diary.
- You can get the first and the last story from your diary.
- You can get any stories from your specified index.
- You can set new story to at the specified index with the specified story.
- You can get index of your specified story.
- You can get all stories from your diary.

Lab#2 DiaryStory



LocalDate

```
import java.time.LocalDate;
import java.time.Month;
import java.time.format.DateTimeFormatter;

public class TestDate {

    public static void main(String[] args) throws Exception {
        System.out.println("LocalDate is A date without a time-zone");
        LocalDate ld1 = LocalDate.now();
        LocalDate ld2 = LocalDate.of(2018, 1, 20);
        LocalDate ld3 = LocalDate.of(2018, Month.JANUARY, 30);
        LocalDate ld4 = LocalDate.of(2018, 10, 31);
        System.out.println("ld1: " + ld1);
        System.out.println("ld2: " + ld2);
        System.out.println("ld3: " + ld3);
        System.out.println("ld4: " + ld4);

        System.out.println("\nFormat Date...");
        DateTimeFormatter sdf1 = DateTimeFormatter.ofPattern("yyyy-MM, dd");
        DateTimeFormatter sdf2 = DateTimeFormatter.ofPattern("yyyy-MM-dd");
        System.out.println("format(ld1) : " + sdf1.format(ld1));
        System.out.println("format(ld2) : " + sdf1.format(ld2));
        System.out.println("format(ld3) : " + sdf2.format(ld3));
        System.out.println("format(ld4) : " + sdf2.format(ld4));
```

<output>
LocalDate is A date without a time-zone.
ld1: 2018-10-31
ld2: 2018-01-20
ld3: 2018-01-30
ld4: 2018-10-31

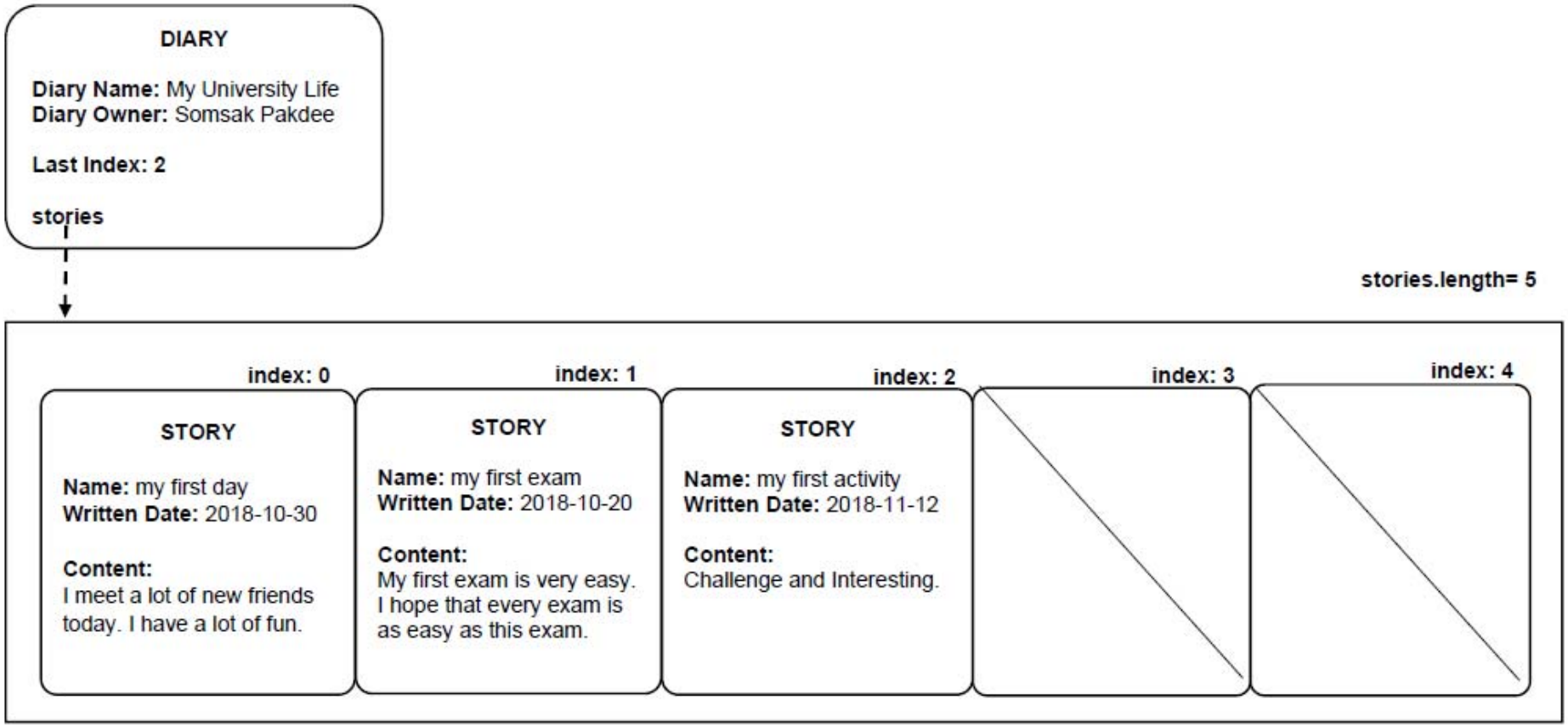
Format Date...
format(ld1) : 2018-10, 31
format(ld2) : 2018-01, 20
format(ld3) : 2018-01-30
format(ld4) : 2018-10-31

```
        System.out.println("\nis before, after, or equals between Two Dates...");
        if (ld2.isBefore(ld1)) {
            System.out.println("Date2 is before Date1");
        }
        if (ld1.isAfter(ld3)) {
            System.out.println("Date1 is after Date3");
        }
        if (ld3.isEqual(ld4)) {
            System.out.println("Date3 is equal Date4");
        }
        if (ld1.isEqual(ld4)) {
            System.out.println("Date1 is equal Date4");
        }

        System.out.println("\nCompare less, greater than, or equals...");
        if (ld1.compareTo(ld2) > 0) {
            System.out.println("Date1 is after Date2");

        } else if (ld1.compareTo(ld2) < 0) {
            System.out.println("Date1 is before Date2");

        } else if (ld1.compareTo(ld2) == 0) {
            System.out.println("Date1 is equal to Date2");

        }
    }
}
```

<output>
is before, after, or equals between Two Dates...
Date2 is before Date1
Date1 is after Date3
Date1 is equal Date4

Compare less, greater than, or equals...
Date1 is after Date2

DiaryStory


DIARY
Diary Name: My University Life
Diary Owner: Somsak Pakdee
Last Index: 2
stories
stories.length= 5
index: 0
STORY
Name: my first day
Written Date: 2018-10-30
Content:
I meet a lot of new friends today. I have a lot of fun.
index: 1
STORY
Name: my first exam
Written Date: 2018-10-20
Content:
My first exam is very easy. I hope that every exam is as easy as this exam.
index: 2
STORY
Name: my first activity
Written Date: 2018-11-12
Content:
Challenge and Interesting.
index: 3
index: 4

Lab#2.1

Look at the Diary/Story Diagram

1. Create **Story Class**, implements *all attributes, constructor, and methods*
2. Create **Diary Class**, implements *all attributes, constructor and only methods*
   - `Diary constructor` with accepting the maximum number of stories parameter to set the story array size and set lastIndex to -1
   - `getDiaryName()`
   - `setDiaryName(String diaryName)`
   - `getDiaryOwner()`
   - `setDiaryOwner(String diaryOwner)`
   - `getLastIndex()`

Lab#2.2

- In the Diary class, overrides ***toString()*** method to display diary name, diary owner, and all stories as shown below:

```
Diary{Diary Name=My Memo, Diary Owner=Suparb Rakdee}
//No Story
```

Lab#2.3

- In the Diary class, implements

```
public boolean addStory(String storyName, LocalDate writtenDate,
String content){ }
```

appends the specified story to the end of the diary, return true if complete, otherwise return false.

- Calling ***toString()*** in the Diary class to print Diary object, you should see your added story.

```
Diary{Diary Name=My Memo, Diary Owner=Suparb Rakdee}

Story{storyName=My First Exam, writtenDate=2018-10-31, content=It is
very easy exam}
```

Lab#2.4

In the Diary class, implements

- `public Story getFirstStory(){ }`
  return the first story from your diary, if no story, return null.

- `public Story getLastStory() { }`
  return the last story from your diary, if no story, return null.

- `public Story getStory(int index) { }`
  return story from your specified index, if no story, return null.

- `public Story setStory(int index, Story s) {`
  replaces the story at the specified position in the diary with the specified story. If index is out of range of the existing stories, return null. Otherwise, return story element previously at the specified index.

Lab#2.5

In the Diary class, implements

- `public int indexOf(Story s) { }`

returns the index of the first occurrence of the specified story in the diary, or -1 if this diary does not contain the story. Your story argument may contain only story name or written date (or both).

To support *indexOf()* method, In the Story class, you should implement *isMatch(Story s)*

- `public boolean isMatch(Story s) { }`

return true if this story name is equal to story name of story argument or

return true if this written date is equal to written date of story argument or

return true if both this story name and written date are equal to story name and written date of story argument. Otherwise, return false

Lab#2.6

- `public Story[] getStories() { }`

  returns an array containing all of the stories in the diary. If no story, return null.

- `public Story removeStory(int removedIndex) { }`

  removes the story at the specified position in the diary. Shift any subsequence stories to the left. returns the element that was removed from the diary. If index is out of range, return null.