



Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”  
Факультет інформатики та обчислювальної техніки  
Кафедра інформаційних систем та технологій

## **Лабораторна робота № 2**

Колекції. Множина HashSet. Асоціативні масиви Map.

Виконали  
студенти групи ІА-23:  
Архип'юк К. О.,  
Богаченко А. В.,  
Кашуб'як С. М.  
Єрмак Д.Р.

Перевірив:  
Колеснік В. М.

## Хід роботи:

### Варіант 4

1. Ознайомитись з javadoc для наступних інтерфейсів, класів та методів:

- Set
- HashSet
- Object.equals(), Object.hashCode()
- Map
- HashMap

2. Виконати завдання лабораторної роботи №10, замінивши списки List (ArrayList та LinkedList) на множини Set (HashSet). Проаналізувати предметну область та на власний розсуд додати функціональність, для реалізації якої використати Map (TreeMap або HashMap).

### Main.java

```
import java.util.*;

public class Main {
    public static void main(String[] args) {
        VideoBlog blog = new VideoBlog("Nanana");
        Video video1 = new Video("Git tutorial",
            "https://youtu.be/UQ8nXehmOro", 56_000, 8300, 500);
        Video video2 = new Video("Student life",
            "https://youtu.be/dQw4w9WgXcQ", 30_000, 4800, 500);
        Video video3 = new Video("My plans for future",
            "https://youtu.be/fMSS6diyXQE", 145_000, 10000, 23);
        VideoBlog.addVideo(video1);
        VideoBlog.addVideo(video2);
        VideoBlog.addVideo(video3);

        Comment com1 = new Comment("I like it!", 123, 15);
        Comment com2 = new Comment("You have such a pretty dog!", 9999, 0);
        Comment com3 = new Comment("You are so lazy. One video per month!
Unsubscribe!!!", 3, 87);
        video1.addComment(com1);
        video1.addComment(com2);
        video2.addComment(com3);
        video3.addComment(com1);
        video3.addComment(com2);
        video3.addComment(com3);

        System.out.println("Total views: " + task1());
        System.out.println("Is there any comment which has more likes than
it`s video: " + task2());
        System.out.print("Video which have the biggest number of dislike: ");
        task3(); System.out.println();
        System.out.println("Videos with more than 50000 views: " + task4());
    }
}
```

```

/*1-типізований
2-for-each
3-нетипізований*/

//1. Сумарна кількість переглядів усіх відео разом
public static int task1() {
    int sumviews = 0;
    int thisviews;
    Iterator<Video> i = VideoBlog.getVideos().iterator();
    while (i.hasNext()) {
        Video video = i.next();
        thisviews = video.getVidviews();
        sumviews += thisviews;
    }
    return sumviews;
}

//2. Чи є коментар, що набрав лайків більше, ніж відео, до якого він
дописаний
static boolean task2() {
    int videolikes;
    int commentlikes;
    for (Video video : VideoBlog.allvideos) {
        videolikes = video.getVidlikes();
        for (Comment comment : Video.comments) {
            commentlikes = comment.getCommentLikes();
            if (commentlikes > videolikes) return true;
        }
    }
    return false;
}

//3. Відео, що набрало найбільшу кількість дізлайків
public static void task3() {
    Set<Video> maxdislike = new HashSet<>();
    int dislike;
    int max = 0;
    for (Iterator i = VideoBlog.getVideos().iterator(); i.hasNext(); ) {
        Video video = (Video) i.next();
        dislike = video.getViddislikes();
        if (dislike > max) max = dislike;
    }
    for (Iterator i = VideoBlog.getVideos().iterator(); i.hasNext(); ) {
        Video video = (Video) i.next();
        dislike = video.getViddislikes();
        if ((dislike == max) & (max != 0)) {
            maxdislike.add(video);
        }
    }
    for (Video n : maxdislike) System.out.print(n.name() + " ");
}

//Відео, що мають більше 50_000 переглядів (реалізація HashMap)
public static HashMap<String, String> task4() {
    HashMap<String, String> vidViewsStats = new HashMap<>();
    for (Iterator<Video> i = VideoBlog.getVideos().iterator();
i.hasNext(); ) {
        Video video = i.next();
        if (video.vidviews >= 50_000) {
            vidViewsStats.put(video.name, video.URL);
        }
    }
    return vidViewsStats;
}

```

```
}  
}
```

## VideoBlog.java

```
import java.util.HashSet;  
import java.util.Set;  
  
public class VideoBlog {  
    private String nickname;  
    static Set<Video> allvideos = new HashSet<>();  
  
    public static void addVideo(Video video) {  
        allvideos.add(video);  
    }  
  
    public static Set<Video> getVideos() {  
        return allvideos;  
    }  
  
    public VideoBlog(String nickname) {  
        this.nickname = nickname;  
    }  
}
```

## Video.java

```
import java.util.HashSet;  
import java.util.Objects;  
import java.util.Set;  
  
public class Video {  
    String name;  
    String URL;  
    int vidviews;  
    int vidlikes;  
    int viddislikes;  
    static Set<Comment> comments = new HashSet<>();  
  
    public Video(String name, String URL, int vidviews, int vidlikes, int  
viddislikes) {  
        this.name = name;  
        this.URL = URL;  
        this.vidviews = vidviews;  
        this.vidlikes = vidlikes;  
        this.viddislikes = viddislikes;  
    }  
  
    public int getVidviews() {  
        return this.vidviews;  
    }  
  
    public int getVidlikes() {  
        return this.vidlikes;  
    }  
  
    public int getViddislikes() {  
        return this.viddislikes;  
    }  
}
```

```

    public static void addComment(Comment comment) {
        comments.add(comment);
    }

    public String name() {
        return this.name;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Video video = (Video) o;
        return Objects.equals(name, video.name) && URL.equals(video.URL);
    }

    @Override
    public int hashCode() {
        return Objects.hash(name, URL);
    }
}

```

## Comment.java

```

public class Comment {
    String commentText;
    int commentLikes;
    int commentDislikes;

    public Comment(String commentText, int commentLikes, int commentDislikes)
    {
        this.commentText = commentText;
        this.commentLikes = commentLikes;
        this.commentDislikes = commentDislikes;
    }

    public int getCommentLikes() {
        return this.commentLikes;
    }
}

```

## Result:

```

Total views: 231000
Is there any comment which has more likes than it's video: true
Video which have the biggest number of dislike: Student life; Git tutorial;
Videos with more than 50000 views: {Git tutorial=https://youtu.be/UQ8nXehm0ro, My plans for future=https://youtu.be/fMSS6diyXQE}

```

**Висновок:** під час виконання лабораторної роботи ми продовжили ознайомлення з Set та HashSet, а також додали функціональність для реалізації якої використали Map.