

תרגול מס' 4: המתרגם

שימוש במחלקות קיימות

משימה

- תכנית המתרגמת קטעי טקסט לשפה אחרת
- הקלט: קובץ המכיל את קטעי הטקסט וכן את השפה אליה רוצים לתרגם



תוכנת מתקדם בשפת Java
אוניברסיטת תל אביב

2

שאלות

- האם כבר יש שירות תרגום שאנחנו יכולים להשתמש בו?
- אולי נוכל לעשות שימוש ב Google Translate?
- כיצד קוראים מקבצים?
- מה הפורמט של הקלט?
- נצטרך להחליט

תוכנת מתקדם בשפת Java
אוניברסיטת תל אביב

3

הפשטה

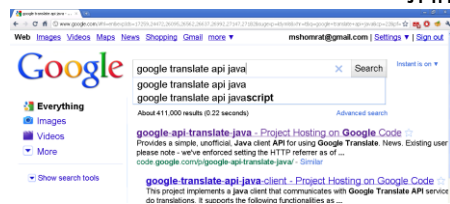
- כצעד ראשון נפתור בעיה הרבה יותר פשוטה
- תכנית שמתרגמת את המילה "Hello" מאנגלית לצרפתית
- יש: שימוש בשירות תרגום
- אין: קלט, טקסט, עבודה עם קבצים, פורמט

תוכנת מתקדם בשפת Java
אוניברסיטת תל אביב

4

API – Application Programming Interface

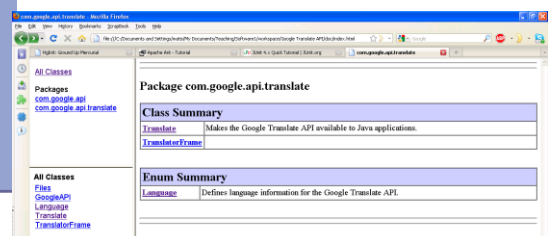
ממשק המאפשר לאפליקציה לתקשר עם תוכנה אחרת



תוכנת מתקדם בשפת Java
אוניברסיטת תל אביב

5

Eureka!



תוכנת מתקדם בשפת Java
אוניברסיטת תל אביב

6

שלב א'

```
public class TranslatorEngine {
    private static final String HTTP_REFERER =
        "http://www.cs.tau.ac.il";

    public static void main(String[] args) throws Exception {
        Translate.setHttpReferrer(HTTP_REFERER);

        String translatedWord = Translate.execute("Hello",
            Language.ENGLISH, Language.FRENCH);

        System.out.println(translatedWord);
    }
}
```

אינטראקציה עם המשתמש

- נתחיל להתקדם עקב בצד אגודל אל היעד שלנו
- קלט מהמשתמש יינתן בשורת הפקודה
 - פרמטר ראשון: המילה לתרגום
 - פרמטר שני: שפת המקור
 - פרמטר שלישי: שפת היעד

שלב ב'

```
public class TranslatorEngine {
    private static final String HTTP_REFERER =
        "http://www.cs.tau.ac.il";

    public static void main(String[] args) throws Exception {
        Translate.setHttpReferrer(HTTP_REFERER);
        System.out.println(
            Translate.execute(args[0],
                Language.fromString(args[1]),
                Language.fromString(args[2])));
    }
}
```

קריאת קלט

- נקרא קלט מהמשתמש (console)
 - עדיין מילה אחת
 - אין שימוש בקבצים
- נשתמש במחלקה Scanner
 - מה הפורמט של הקלט?

המחלקה Scanner

- סורק טקסט פשוט
- "שובר" את הקלט לרכיביו השונים (מילה, מספר וכדומה)
- בעת יצירה מקבל כפרמטר מהיכן לקרוא את הקלט

```
Scanner s = new Scanner(System.in);
int aInt = s.nextInt();
float aFloat = s.nextFloat();
String aString = s.next();
String aLine = s.nextLine();
```

פורמט הקלט

- מהו הפרוטוקול המשותף שחולקים האפליקציה והמשתמש לצורך התקשורת ביניהם
 - איזה מידע דרוש
 - כיצד הוא מקודד (מספר, מחרוזת, ...)
 - מה סדר הפרמטרים
- נבחר: `<word> <source-lang> <target-lang>`

דוגמא

```
Scanner s = new Scanner(System.in);
System.out.println("enter line:");
while (s.hasNext())
    System.out.println(s.next());
```

קרא מ- standard input

קרא את ה-Token הבא

שלב ג'

```
public class TranslatorEngine {
    private static final String HTTP_REFERER =
        "http://www.cs.tau.ac.il";

    public static void main(String[] args) throws Exception {
        Translate.setHttpReferrer(HTTP_REFERER);

        Scanner s = new Scanner(System.in);
        String[] fragments = s.nextLine().split(" ");
        System.out.println(Translate.execute(fragments[0],
            Language.fromString(fragments[1]),
            Language.fromString(fragments[2])));
    }
}
```

קבצים

- במקום לקרוא את שורת הקלט מהמשתמש נקרא אותה מקובץ
 - קובץ מיצג ע"י המחלקה File
 - נאתחל את האובייקט עם המסלול (path) לקובץ
- ```
File f =
 new File("C:\\Software1\\example.txt");
```

## שלב ד'

```
public class TranslatorEngine {
 private static final String HTTP_REFERER =
 "http://www.cs.tau.ac.il";
 private static final String FILE_NAME = ...

 public static void main(String[] args) throws Exception {
 Translate.setHttpReferrer(HTTP_REFERER);

 Scanner s = new Scanner(new File(FILE_NAME));
 String[] fragments = s.nextLine().split(" ");
 System.out.println(Translate.execute(fragments[0],
 Language.fromString(fragments[1]),
 Language.fromString(fragments[2])));
 }
}
```

## קלטים מרובים

- מספר שורות קלט מקובץ
  - נקרא מספר קלטים עד לסוף הקובץ
- שימוש במתודות hasNextLine ו-nextLine

## שלב ה'

```
public class TranslatorEngine {
 private static final String HTTP_REFERER =
 "http://www.cs.tau.ac.il";
 private static final String FILE_NAME = ...

 public static void main(String[] args) throws Exception {
 Translate.setHttpReferrer(HTTP_REFERER);

 Scanner s = new Scanner(new File(FILE_NAME));
 while (s.hasNextLine()) {
 String[] fragments = s.nextLine().split(" ");
 System.out.println(Translate.execute(fragments[0],
 Language.fromString(fragments[1]),
 Language.fromString(fragments[2])));
 }
 }
}
```

## פיסקה

- פיסקה ולא רק מילה אחת
- מה יהיה הפורמט החדש?

■ נבחר:

<source-lang>#<target-lang>#<paragraph>

## Example - Scanner Set delimiters

```
String input = "1 fish 2 fish red fish blue fish";
Scanner s =
 new Scanner(input).useDelimiter(" *fish *");
while (s.hasNext())
 System.out.println(s.next());
s.close();
```

## שלב ו'

```
public class TranslatorEngine {
 private static final String HTTP_REFERERER = ...
 private static final String FILE_NAME = ...

 public static void main(String[] args) throws Exception {
 Translate.setHttpReferrer(HTTP_REFERERER);

 Scanner s = new Scanner(new File(FILE_NAME));
 s.useDelimiter("#");
 Language from = Language.fromString(s.next());
 Language to = Language.fromString(s.next());
 s.skip("#");
 StringBuffer text = new StringBuffer();
 while (s.hasNextLine()) {
 text.append(s.nextLine());
 text.append(' ');
 }
 System.out.println(Translate.execute(text.toString(),
 from, to));
 }
}
```

## לאן עכשיו?

- תרגום מספר קבצים
- מספר פסקאות בקובץ יחיד
- לוותר על חלק מהפרמטרים
- זיהוי אוטומטי של שפת הקלט
- שימוש בשמות משמעותיים לשפות
- GUI
- ...

## מה חסר?

- טיפול בשגיאות
- חסרים פרמטרים
- שפות לא מוגדרות
- מנגנון התרגום נכשל
- קובץ לא קיים
- ...