

## תרגיל פונקציות, קבצים, מודולים

דוגמת הרצה אחרי כל התרגילים:

```
# main.py

from crud import create, read, update, delete

create(name="tal", mail="aa@aa.com", grade="90")
create(name="gal", mail="gg@gg.com", grade="80")
print(read())
update(old_value="tal", new_value="yosi")
print(read())
delete(value="gal")
print(read())
```

name	grade	mail
tal	90	aa@aa.com
gal	80	gg@gg.com

  

name	grade	mail
yosi	90	aa@aa.com
gal	80	gg@gg.com

  

name	grade	mail
yosi	90	aa@aa.com
	80	gg@gg.com

1. כתבו פונקציות ליצור, להדפיס, לעדכן ולמחוק ערכים ממילון. ערכי המילון הם שם, מייל וציון של תלמיד:

```
student={"name":"tal", "mail":"tal@tal.com", "grade":100}
```

פתרון:

```
student={"name":"tal", "mail":"tal@tal.com", "grade":100}
```

```
def create(name:str="name", mail:str="mail", grade:int=0):
    global student
    student.update({"name":name, "grade":grade, "mail":mail})
```

```
def read(key:str="name"):
    global student
    print(student.get(key, "no such attribute"))
```

```
def update(item:dict={}):
    global student
    student.update(item)
```

```
def delete(key:str="name"):
    global student
    student[key]=None
```

2. שמרו את המילון בקובץ CSV. יש לכתוב פונקציית עזר שמקבלת את המילון והופכת אותו לשורת ערכים מופרדת בפסיקים.

פתרון:

```
def dict_to_csv(data:dict={}):
    return ",".join(data.values())

def save_item(item:dict={}, filename="students.csv"):
    with open(filename, 'a') as f:
        f.write(dict_to_csv(item))
```

נשנה את הפונקציה create בהתאם:

```
def create(name:str="name", mail:str="mail", grade:int=0):
    save_item({"name":name, "grade":grade, "mail":mail})
```

3. שמרו את המילון ברשימה, כך שתהיה טבלה – רשימה של תלמידים. שמרו את הטבלה בקובץ CSV ע"י שימוש וכתובת פונקציות בהתאם.

פתרון:

```
students=[]

def save_table():
    for student in students:
        save_item(student)
```

4. במקום משתנה גלובלי, כתבו פונקציה הטוענת את המילונים מקובץ. הפונקציה תקבל את פורמט הקובץ כ named argument ותפתח אותו בהתאם. שנו את read כך שתחזיר את הטבלה כמחרוזת מהקובץ.

```
def load_data(filename="student.csv"):
    students=[]
    with open(filename) as f:
        for line in f.readlines():
            students.append(dict(zip(["name", "mail", "grade"], line.split(','))))
    return students

def read():
    students=load_data()
    students_string=""
    print("name\tgrade\tmail")
    for student in students:
        students_string+="\t".join(student.values())+"\n"
```

```
return students_string
```

5. בונוס: שפרו את update | delete כך שיעדכנו את הנתונים בקובץ.

פתרון:

```
def update(key:str="name", old_value:str="", new_value:str=""):
    students=load_data()
    updated_students=[]
    for student in students:
        if student[key]==old_value:
            student[key]=new_value
            updated_students.append(student)
    save_table(table=updated_students)
```

```
def delete(key:str="name", value=""):
    update(key=key, old_value=value, new_value="")
```

6. הפרידו את הפונקציות למודולים: main.py, crud.py, files.py

```
# crud.py
```

```
from files import save_item, load_data, save_table
```

```
def create(name:str="name", mail:str="mail", grade:str="0"):
    save_item({"name":name, "grade":grade, "mail":mail})
```

```
def read():
    students=load_data()
    students_string=""
    print("name\tmail\tgrade")
    for student in students:
        students_string+="\t".join(student.values())+"\n"
    return students_string
```

```
def update(key:str="name", old_value:str="", new_value:str=""):
    students=load_data()
    updated_students=[]
    for student in students:
        if student[key]==old_value:
            student[key]=new_value
            updated_students.append(student)
    save_table(table=updated_students)
```

```
def delete(key:str="name", value=""):
    update(key=key, old_value=value, new_value="")
```

```
# files.py
```

```
import crud
```

```
def dict_to_csv(data:dict={}):  
    return ",".join(data.values())
```

```
def save_item(item:dict={}, filename="students.csv"):  
    with open(filename, 'a') as f:  
        f.write(dict_to_csv(item)+"\n")
```

```
def load_data(filename:str="students.csv"):  
    students=[]  
    with open(filename) as f:  
        for line in f.readlines():  
            students.append(dict(zip(["name", "mail", "grade"], line.replace(' ', '').strip('\n').split(','))))  
    return students
```

```
def save_table(filename:str="students.csv", table:list=[]):  
    with open(filename, 'w') as f:  
        for student in table:  
            f.write(dict_to_csv(student)+"\n")
```

```
# main.py
```

```
from crud import create, read, update
```

```
create(name="tal", mail="aa@aa.com", grade="90")  
create(name="gal", mail="gg@gg.com", grade="80")  
print(read())  
update(old_value="tal", new_value="yosi")  
print(read())
```