



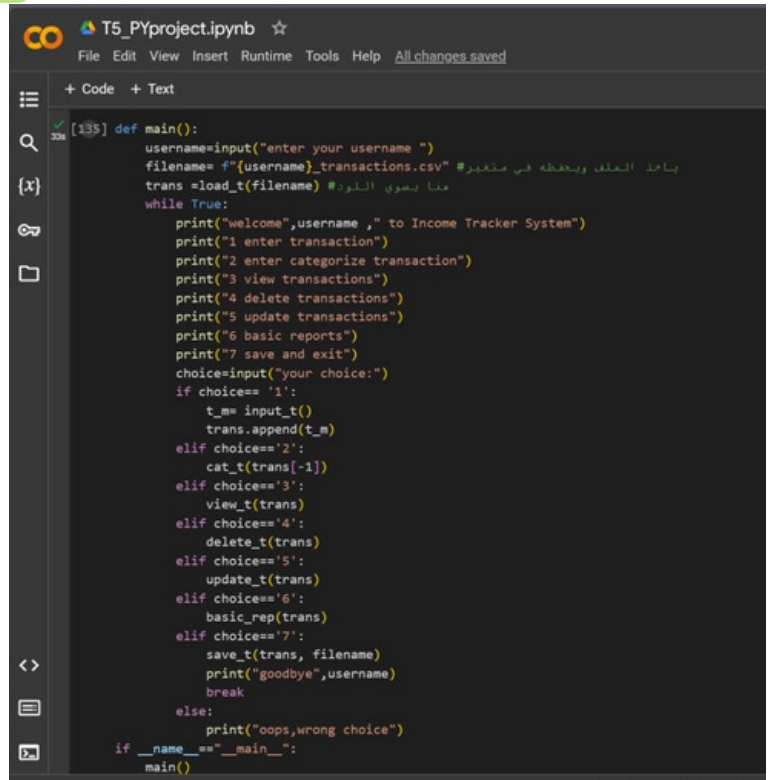
# INCOME TRACKER

MADAWI BIN SHAWIAH  
&  
KHAFUQ ALTHEYAB

**MAIN FUN:** To call the functions

## Code:

```
def main():
    username=input("enter your username ")
    filename= f"{username}_transactions.csv" #ياخذ الملف ويحفظه في متغير
    trans =load_t(filename) #هنا يسوي اللود
    while True:
        print("welcome",username ," to Income Tracker System")
        print("1 enter transaction")
        print("2 enter categorize transaction")
        print("3 view transactions")
        print("4 delete transactions")
        print("5 update transactions")
        print("6 basic reports")
        print("7 save and exit")
        choice=input("your choice:")
        if choice== '1':
            t_m= input_t()
            trans.append(t_m)
        elif choice=='2':
            cat_t(trans[-1])
        elif choice=='3':
            view_t(trans)
        elif choice=='4':
            delete_t(trans)
        elif choice=='5':
            update_t(trans)
        elif choice=='6':
            basic_rep(trans)
        elif choice=='7':
            save_t(trans, filename)
            print("goodbye",username)
            break
        else:
            print("oops,wrong choice")
    if name=="main":
        main()
```



The screenshot shows a Jupyter Notebook titled 'T5\_PYproject.ipynb'. The code is written in Python and matches the code block provided. The notebook interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help) and a toolbar with icons for running, saving, and other actions. The code is displayed in a dark-themed editor with syntax highlighting.

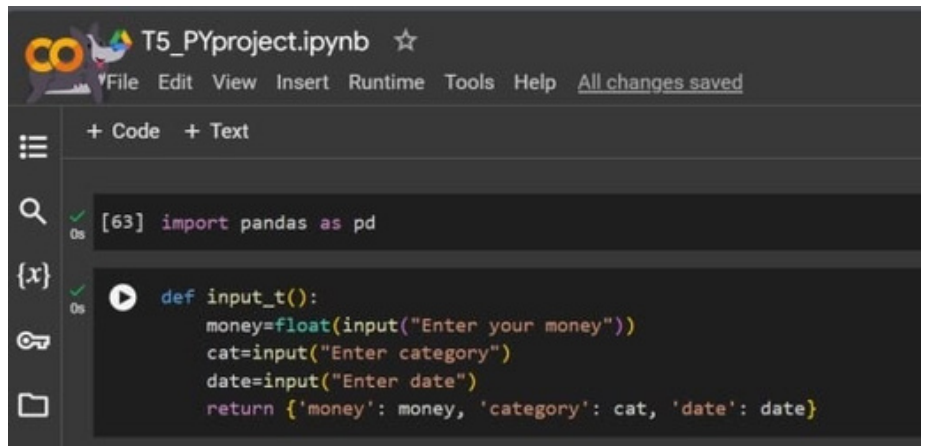
## Output:

```
⊗ Enter your username:sarah
welcome sarah to Income Tracker System
1 enter transaction
2 enter categorize transaction
3 view transactions
4 delete transactions
5 update transactions
6 basic reports
7 save and exit
your choice:
```

## INCOM ENTRY FEATURE:

### Code:

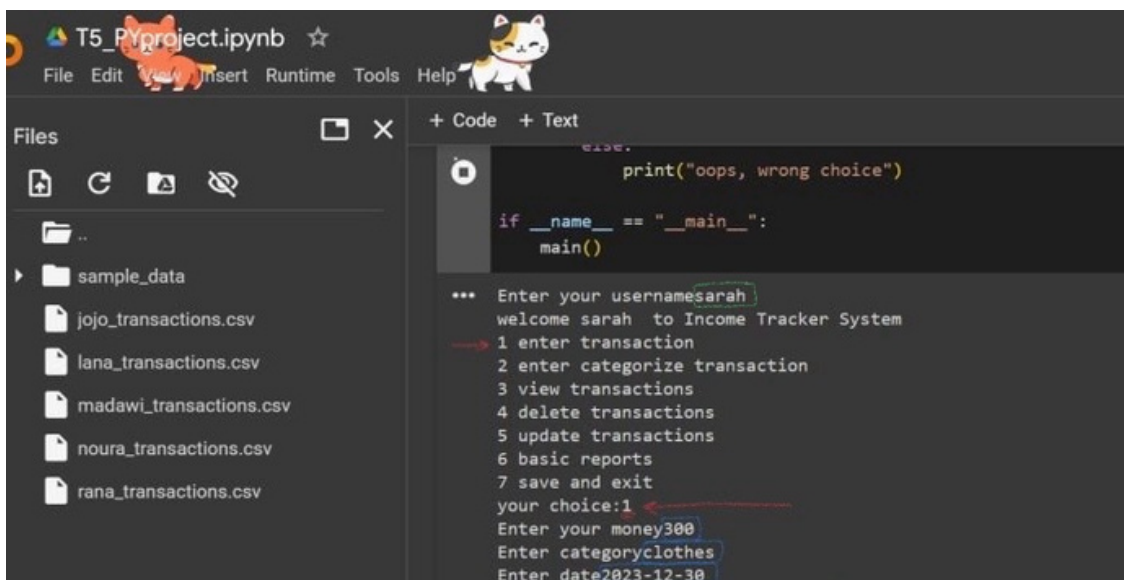
```
def input_t():  
    money=float(input("Enter your money"))  
    cat=input("Enter category")  
    date=input("Enter date")  
    return {'money': money, 'category': cat, 'date': date}
```



The screenshot shows a Jupyter Notebook titled 'T5\_PYproject.ipynb'. The code cell contains the following Python code:

```
[63] import pandas as pd  
  
def input_t():  
    money=float(input("Enter your money"))  
    cat=input("Enter category")  
    date=input("Enter date")  
    return {'money': money, 'category': cat, 'date': date}
```

### Output:



The screenshot shows the Jupyter Notebook interface with the 'Files' sidebar on the left. The code cell contains the following Python code:

```
else:  
    print("oops, wrong choice")  
  
if __name__ == "__main__":  
    main()
```

The output of the code is displayed below the code cell:

```
*** Enter your username:sarah  
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:1  
Enter your money:300  
Enter category:clothes  
Enter date:2023-12-30
```

## CATEGORISATION FEATURE:

### Code:

```
def cat_t(tran):  
    cate=input("enter category") #استقبال كاتقري  
    tran['category']=cate #إضافة
```

```
✓ [121] def cat_t(tran):  
0s      cate=input("enter category") #استقبال كاتقري  
      tran['category']=cate #إضافة
```

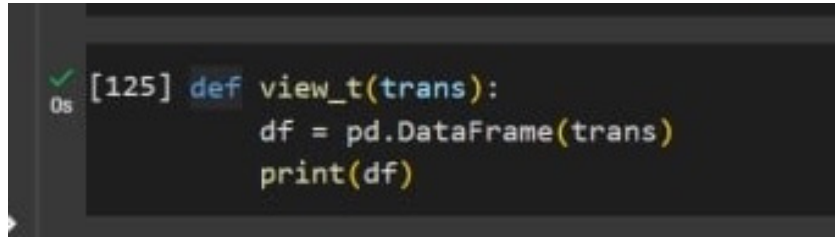
### Output:

```
*** welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:2  
enter categoryshopping  
welcome sarah to Income Tracker System
```

## VIEW FEATURE:

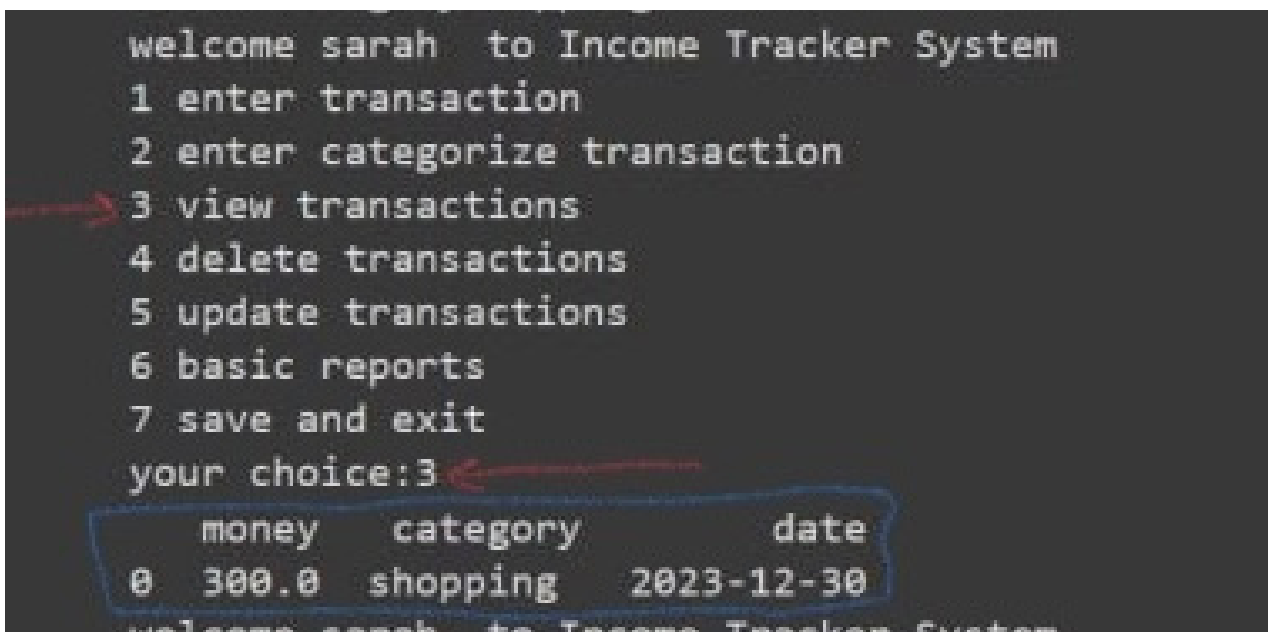
### Code:

```
def view_t(trans):  
    df = pd.DataFrame(trans)  
    print(df)
```



```
[125] def view_t(trans):  
      df = pd.DataFrame(trans)  
      print(df)
```

### Output:



```
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
→ 3 view transactions  
4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:3  


|   | money | category | date       |
|---|-------|----------|------------|
| 0 | 300.0 | shopping | 2023-12-30 |

  
welcome sarah to Income Tracker System
```

## DELETE FEATURE:

### Code:

```
def delete_t(trans):  
    view_t(trans)  
    d = int(input("enter the index of the transaction"))  
    if d < 0 or d >= len(trans): # يشيك على العملية هل هل موجوده ولا لا  
        print("NOT corect index")  
        return  
    del_t = trans.pop(d)  
    print("done")
```

```
✓ [115] def delete_t(trans):  
    view_t(trans)  
    d = int(input("enter the index of the transaction"))  
    if d < 0 or d >= len(trans): # يشيك على العملية هل هل موجوده ولا لا  
        print("NOT corect index")  
        return  
    del_t = trans.pop(d)  
    print("done")
```

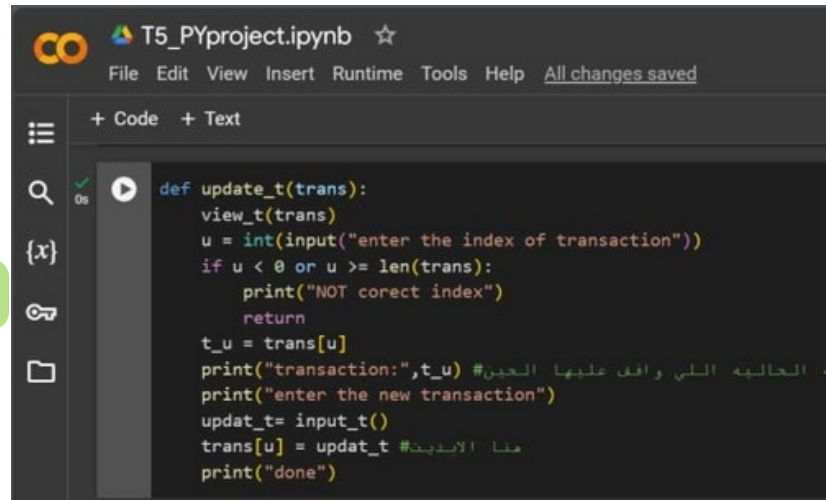
### Output:

```
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
→ 4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:4  
    money    category    date  
0  300.0  shopping  2023-12-30  
enter the index of the transaction0  
done ✓
```

## UPDATE FEATURE:

### Code:

```
def update_t(trans):  
    view_t(trans)  
    u = int(input("enter the index of transaction"))  
    if u < 0 or u >= len(trans):  
        print("NOT corect index")  
        return  
    t_u = trans[u]  
    print("transaction:",t_u)  
    print("enter the new transaction")  
    updat_t= input_t()  
    trans[u] = updat_t # هنا الابدیت  
    print("done")
```



The screenshot shows a Jupyter Notebook interface with a file named 'T5\_PYproject.ipynb'. The code editor displays the 'update\_t' function, which includes a call to 'view\_t(trans)', prompts for a transaction index, validates it, and then updates the transaction at that index with a new value entered by the user. The code is written in Python and includes Arabic comments.

### Output:

```
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:1  
Enter your money9000  
Enter categoryincome  
Enter date2024-2-27  
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
5 update transactions  
6 basic reports  
7 save and exit  
your choice:
```

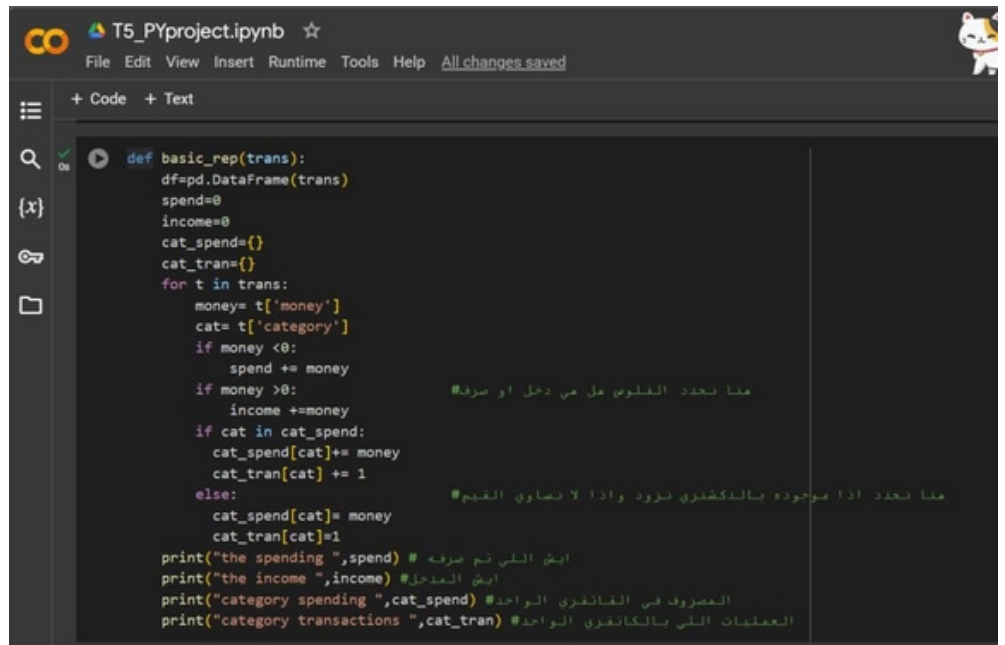
A new transaction has been added to apply the update feature, because all previous transactions have been deleted

```
welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
→ 5 update transactions  
6 basic reports  
7 save and exit  
your choice:5  
money category date  
0 9000.0 income 2024-2-27  
enter the index of transaction0  
transaction: {'money': 9000.0, 'category': 'income', 'date': '2024-2-27'}  
enter the new transaction  
Enter your money10500  
Enter categoryincome  
Enter date2024-3-1  
done
```

## BASIC REPORTS FEATURE:

### Code:

```
def basic_rep(trans):  
    df=pd.DataFrame(trans)  
    spend=0  
    income=0  
    cat_spend={}  
    cat_tran={}  
    for t in trans:  
        money= t['money']  
        cat= t['category']  
        if money <0:  
            spend += money  
        if money >0:  
            income +=money  
        if cat in cat_spend:  
            cat_spend[cat]+= money  
            cat_tran[cat] += 1  
        else:  
            cat_spend[cat]= money  
            cat_tran[cat]=1  
    print("the spending ",spend)  
    print("the income ",income)  
    print("category spending ",cat_spend)  
    print("category transactions ",cat_tran)
```



```
T5_PYproject.ipynb ☆  
File Edit View Insert Runtime Tools Help All changes saved  
+ Code + Text  
def basic_rep(trans):  
    df=pd.DataFrame(trans)  
    spend=0  
    income=0  
    cat_spend={}  
    cat_tran={}  
    for t in trans:  
        money= t['money']  
        cat= t['category']  
        if money <0:  
            spend += money  
        if money >0:  
            income +=money  
        if cat in cat_spend:  
            cat_spend[cat]+= money  
            cat_tran[cat] += 1  
        else:  
            cat_spend[cat]= money  
            cat_tran[cat]=1  
    print("the spending ",spend) # ايش اللي تم صرفه  
    print("the income ",income) # ايش المدخل  
    print("category spending ",cat_spend) # المصروف في القاتري الواحد  
    print("category transactions ",cat_tran) # العمليات اللي بالكاتري الواحد
```

### Output:

```
👉 welcome sarah to Income Tracker System  
1 enter transaction  
2 enter categorize transaction  
3 view transactions  
4 delete transactions  
5 update transactions  
→ 6 basic reports  
7 save and exit  
your choice:6  
the spending 0  
the income 10500.0  
category spending {'income': 10500.0}  
category transactions {'income': 1}
```



# DATA PERSISTENCE FEATURE:

```
def save_t(trans, filename):  
    df = pd.DataFrame(trans)  
    df.to_csv(filename)
```

```
def load_t(filename):  
    try:  
        df = pd.read_csv(filename)  
        trans = df.to_dict('records')  
        return trans  
    except FileNotFoundError:  
        return []
```

```
[126] def save_t(trans, filename):  
        df = pd.DataFrame(trans)  
        df.to_csv(filename) # هنا يحفظ العمليات في ملف csv
```

```
T5_PYproject.ipynb  
File Edit View Insert Runtime Tools Help All changes saved  
+ Code + Text  
def load_t(filename): # هنا بقرا الملف عشان نأخذ منه العملية  
    try:  
        df = pd.read_csv(filename)  
        trans = df.to_dict('records')  
        return trans  
    except FileNotFoundError:  
        return [] # هنا لو جاني البيوت فاضي راج يرجع لسته فاضي
```

