

SIMPLE CURRENCY CONVERTER

ارایه شده توسط محصلین صنف اول مخابرات
دیپارتمانت Information Science Engineering
سجاد اکبری، یاسین رحیمی، یحیی حمیدی، و حیدرالله رحیمی.

```
import tkinter as tk
from tkinter import ttk

class SimpleCurrencyConverter:
    def __init__(self, root):
        self.root = root
        self.root.title("بدل ارز ساده")
        self.root.geometry("400x250")

        # ثابت نرخهای
        self.rates = {
            'USD': 1.0,
            'EUR': 0.93,
            'IRR': 42000.0,
            'AFG': 66,
            'KDP': 35.0
        }

        self.create_widgets()

    def create_widgets(self):
        tk.Label(self.root, text="بدل ارز", font=("Arial", 16, "bold")).pack(pady=10)

        # فریم ورودی
        input_frame = tk.Frame(self.root)
        input_frame.pack(pady=10)

        tk.Label(input_frame, text="مبلغ:").grid(row=0, column=0, padx=5)
        self.amount_entry = tk.Entry(input_frame, width=15)
        self.amount_entry.grid(row=0, column=1, padx=5)

        # فریم انتخاب ارزی
        currency_frame = tk.Frame(self.root)
        currency_frame.pack(pady=10)

        tk.Label(currency_frame, text="از:").grid(row=0, column=0, padx=5)
        self.from_currency = ttk.Combobox(currency_frame,
                                         values=list(self.rates.keys()))
        self.from_currency.set('USD')
        self.from_currency.grid(row=0, column=1, padx=5)
```

SIMPLE CURRENCY CONVERTER

```
tk.Label(currency_frame, text="﷼:").grid(row=1, column=0, padx=5, pady=5)
self.to_currency = ttk.Combobox(currency_frame,
values=list(self.rates.keys()))
self.to_currency.set('IRR')
self.to_currency.grid(row=1, column=1, padx=5, pady=5)

# دکمه تبدیل
tk.Button(self.root, text="تبدیل", command=self.convert).pack(pady=10)

# نتیجه
self.result_label = tk.Label(self.root, text="", font=("Arial", 12))
self.result_label.pack()

def convert(self):
    try:
        amount = float(self.amount_entry.get())
        from_curr = self.from_currency.get()
        to_curr = self.to_currency.get()

        result = amount * (self.rates[to_curr] / self.rates[from_curr])
        self.result_label.config(text=f"{amount} {from_curr} = {result:.2f} {to_curr}")

    except:
        self.result_label.config(text="خطا در تبدیل")

if __name__ == "__main__":
    root = tk.Tk()
    app = SimpleCurrencyConverter(root)
    root.mainloop()
```

در بالا یک پروژه ساده تبدیل ارز که توسط پایتون کد نویسی شده است را میبینید.

- ❖ در ابتدا از کتابخانه Tkinter برای ساخت رابط کاربری (Interface) استفاده شده است.
- ❖ بعدا برای درج نرخ ها از دیکشنری استفاده شده شده.
- ❖ و با استفاده از متود های Label برچسب(Label) های Interface دیزاین شده.
- ❖ برنامه دارای فریم ورودی است تا از کاربر Data را دریافت کند
- ❖ برنامه دارای لیست کشویی برای انتخاب ارز مورد نظر است.
- ❖ و از متود Try و Except برای تبدیل ارز و جلوگیری از خطأ استفاده شده.

SIMPLE CURRENCY CONVERTER

The image shows two side-by-side screenshots of the Visual Studio Code (VS Code) interface, displaying Python code for a "Simple Currency Converter".

Top Screenshot: This screenshot shows the initial setup of the application. The code defines a class `SimpleCurrencyConverter` that initializes a Tkinter root window, sets its title to "سیمپل ارز تبدیل", and specifies its geometry as 400x250 pixels. It then creates a label and an input frame. A currency frame is created and packed into the root window. The code uses several placeholder strings in English and Persian.

```
import tkinter as tk
from tkinter import ttk

class SimpleCurrencyConverter:
    def __init__(self, root):
        self.root = root
        self.root.title("سیمپل ارز تبدیل")
        self.root.geometry("400x250")

    # ... (remaining code)
```

Bottom Screenshot: This screenshot shows the completed application. The code has been modified to handle user input and conversion logic. It includes a `create_widgets` method to set up the UI, a `convert` method to perform the conversion, and a `__main__` block to run the application. The UI now includes a result label and a conversion button.

```
# ... (initial code)
```

```
def create_widgets(self):
    # ... (initial code)
```

```
    tk.Button(self.root, text="تبدیل", command=self.convert).pack(pady=10)

    # نتیجه نمایش
    self.result_label = tk.Label(self.root, text="", font=("Arial", 12))
    self.result_label.pack()

def convert(self):
    try:
        amount = float(self.amount_entry.get())
        from_curr = self.from_currency.get()
        to_curr = self.to_currency.get()

        result = amount * (self.rates[from_curr] / self.rates[to_curr])
        self.result_label.config(text=f"{amount} {from_curr} = {result:.2f} {to_curr}")
    except:
        self.result_label.config(text="لطفاً از این فیلد های را پر کنید")

if __name__ == "__main__":
    root = tk.Tk()
    app = SimpleCurrencyConverter(root)
    root.mainloop()
```

کد های برنامه

SIMPLE CURRENCY CONVERTER

برنامه Interface

```
C:\> Users > akbari > Desktop > currency converter.py > SimpleCurrencyConverter > create_widgets
1 import tkinter as tk
2 from tkinter import ttk
3
4 class SimpleCurrencyConverter:
5     def __init__(self, root):
6         self.root = root
7         self.root.title("مبدل ارز ساده")
8         self.root.geometry("400x250")
9
10    # نرخهای ثابت
11    self.rates = {
12        'USD': 1.0,
13        'EUR': 0.93,
14        'IRR': 42000.0,
15        'AFG': 66,
16        'KDP': 35.0
17    }
18
19    self.create_widgets()
20
21    def create_widgets(self):
22        tk.Label(self.root, text="مبدل ارز", font=("Arial", 16, "bold")).pack(pady=10)
23
24        # فریم ورودی
25        input_frame = tk.Frame(self.root)
26        input_frame.pack(pady=10)
27
28        tk.Label(input_frame, text="مبلغ:").grid(row=0, column=0, padx=5)
29        self.amount_entry = tk.Entry(input_frame, width=15)
30        self.amount_entry.grid(row=0, column=1, padx=5)
31
32        # فریم انتخاب ارز
33        currency_frame = tk.Frame(self.root)
34        currency_frame.pack(pady=10)
35
36        tk.Label(currency_frame, text="j:").grid(row=0, column=0, padx=5)
37        self.from_currency = ttk.Combobox(currency_frame, values=list(self.rates.keys()))
38
```



لیست کشویی برای انتخاب ارز مبدا و مقصد