



Московский институт электроники
и математики им. А.Н. Тихонова

Прикладная математика

Конвертер валют

Моисеев Артём Романович
Лебедев Матвей Олегович
БПМ225



Постановка задачи

Создайте программу, которая конвертирует валюту из одного номинала в другой, используя фиксированный курс конвертации или API, например Fixer или OpenExchangeRates. Вы можете использовать библиотеку, например tkinter, для создания простого графического пользовательского интерфейса.



Используемые библиотеки



tkinter

Tkinter (от *Tk interface*) — кросс-платформенная графическая библиотека, написанная Стином Лумхольтом и Гвидо ван Россумом. Входит в стандартную библиотеку Python. Tkinter — это удобный интерфейс для работы со средствами Tk. Приложения, созданные на основе этой библиотеки, кросс-платформенные, то есть могут запускаться на разных операционных системах.



Requests

Библиотека requests является стандартным инструментом для составления HTTP-запросов в Python. Python Requests — это библиотека, которая создана для быстрой и простой работы с запросами. Стандартные HTTP-библиотеки Python, например та же Urllib3, часто требуют значительно больше кода для выполнения одного и того же действия, а это затрудняет работу. Requests удобнее и проще.





API Fixer



Fixer.io

Данный сервис поддерживает 170 мировых валют. Данные об обменных курсах мировых валют в режиме реального времени, обновляются каждые 60 секунд (в платном тарифе, в бесплатном - каждый час).

Данные о валютах, представляемых Fixer, поступают от поставщиков финансовых данных и банков, включая Европейский центральный банк.





Используемые функции



Функции пользовательского выбора и ввода данных

Три данные функции запускаются при нажатии пользователем на кнопки выбора исходной валюты, целевой валюты и числа

```
def choose_from_currency():  
    app = ListApp("FROM")  
    app.resizable(True, False)  
    app.mainloop()  
  
def choose_to_currency():  
    app = ListApp("TO")  
    app.resizable(True, False)  
    app.mainloop()  
  
def enter_amount():  
    app = AmountApp()  
    app.resizable(True, False)  
    app.mainloop()
```



Функции управления

Две представленные функции запускаются при нажатии пользователем на кнопку сброса значений и закрытия программы соответственно

```
def clear_entries():  
    global from_currency, to_currency, amount, result_label  
    from_currency = None  
    to_currency = None  
    amount = None  
    btn1.config(text="FROM", fg="#fff")  
    btn2.config(text="TO", fg="#fff")  
    btn3.config(text="AMOUNT", fg="#fff")  
    result_label.config(text="Waiting for conversion...")  
    root.update()  
  
def finish_program():  
    root.destroy()
```

Основная функция (fixer.io)

Основная функция, взятая с сайта API. Выполняет главную задачу - перевод из одной валюты в другую.

```
def currency_conversion():
    if from_currency is None or to_currency is None or (amount is None or (amount < 0.0)):
        return result_label.config(text='ERROR', width=40, height=3,
                                    bg="#363636", fg="#ffc35b", font=("Helvetica", 20))
    url = f"http://data.fixer.io/api/latest?access_key=67188d5aac8d3181c9b6e59118c3af17"
    response = requests.get(url)
    rate = response.json()["rates"][from_currency]
    amount_koeff = (amount) / (rate)
    converted_amount = amount_koeff * response.json()["rates"][to_currency]
    converted_amount = round(converted_amount, 2)
    print_(from_currency, to_currency, amount, converted_amount)
    result_label.config(text='Result: ' + str(converted_amount) + ' ' + str(to_currency),
                        width=40, height=3, bg="#363636", fg="#ffc35b", font=("Helvetica", 20))

    root.update()
```



Функция выбора валют

В результате работы этой функции появляется окно FROM (или TO), в котором пользователь выбирать необходимые валюты из предоставленного списка

```
def __init__(self, type):
    super().__init__()
    self.type = type
    self.title(f"{type}")
    self.label = tk.Label(self,
                           text=f"Choose the currency you want to convert {type}",
                           font=("Helvetica", 17), bg="#363636", fg="#ffc35b",
                           anchor="n")
    self.list = tk.Listbox(self,
                           width=30, height=20, bg="#363636", fg="#ffff",
                           selectbackground="#ffc35b",
                           font=("Helvetica", 15))
    self.print_btn = tk.Button(self, text="CONFIRM",
                               width=15, height=2, bg="#363636", fg="#ffff",
                               font=("Helvetica", 17),
                               command=self.print_selection)

    self.label.pack(fill=tk.BOTH)
    self.list.pack(fill=tk.BOTH)
    self.print_btn.pack(fill=tk.BOTH)

    response = requests.get('http://data.fixer.io/api/latest?access_key=67188d5aac8d3181c9b6e59118c3af17')
    if response.status_code == 200:
        data = response.json()
        rates = data['rates']
        for rate in rates:
            self.list.insert(tk.END, rate)
```



Функция выбора валют

Функция сохраняет выбранное значение валюты и сохраняет его в соответствующую переменную. Также она меняет название кнопки на выбранное значение.

```
def print_selection(self):
    selection = self.list.curselection()
    currency = self.list.get(selection)
    print(currency)
    if self.type == "FROM":
        global from_currency
        from_currency = currency
        btn1.config(text=currency, fg="#ffc35b")
    elif self.type == "TO":
        global to_currency
        to_currency = currency
        btn2.config(text=currency, fg="#ffc35b")
    self.destroy()
```



Функции ввода суммы для конвертирования

После нажатия на кнопку AMOUNT появляется новое окно с полем ввода суммы для конвертирования. Введенное значение сохраняется в переменную и появляется на кнопке как новое название. После ввода значения окно исчезает.

```
def __init__(self):
    super().__init__()
    self.title("AMOUNT")
    self.label = tk.Label(self,
                           text="Enter the amount you want to convert",
                           font=("Helvetica", 17), bg="#363636", fg="#ffc35b",
                           anchor="n")
    self.amount_entry = tk.Entry(self, font=("Helvetica", 15), bg="#363636", fg="#fff")
    self.print_btn = tk.Button(self, text="CONFIRM", font=("Helvetica", 17), bg="#363636", fg="#fff",
                               command=self.print_amount)

    self.label.pack(fill=tk.BOTH)
    self.amount_entry.pack(fill=tk.BOTH)
    self.print_btn.pack(fill=tk.BOTH)
```

```
def print_amount(self):
    global amount
    amount = float(self.amount_entry.get())

    print(amount)
    btn3.config(text=str(amount), fg="#ffc35b")
    self.destroy()
```



Некоторые тесты



Пример работы программы

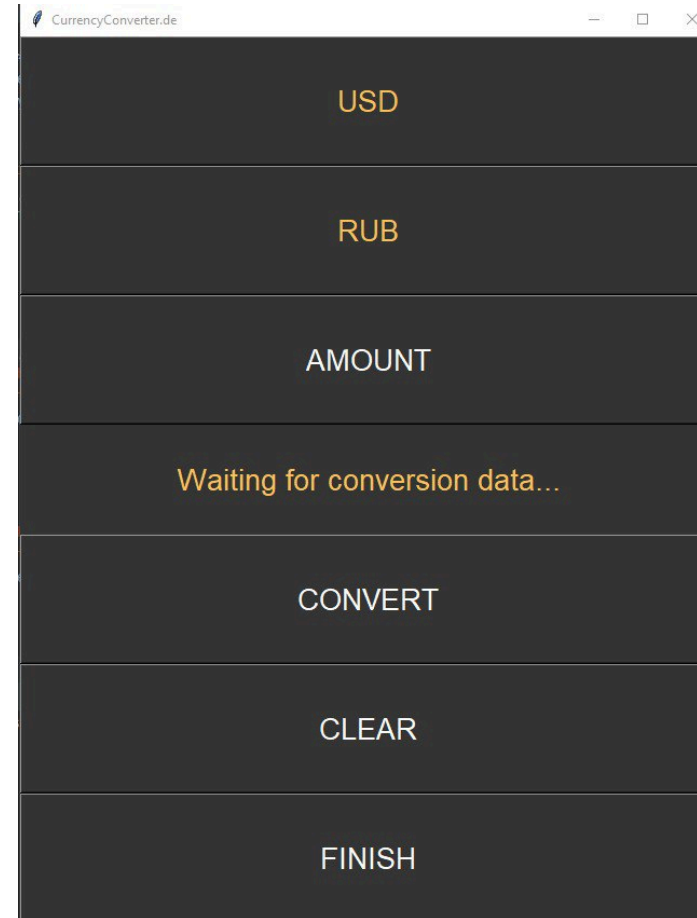
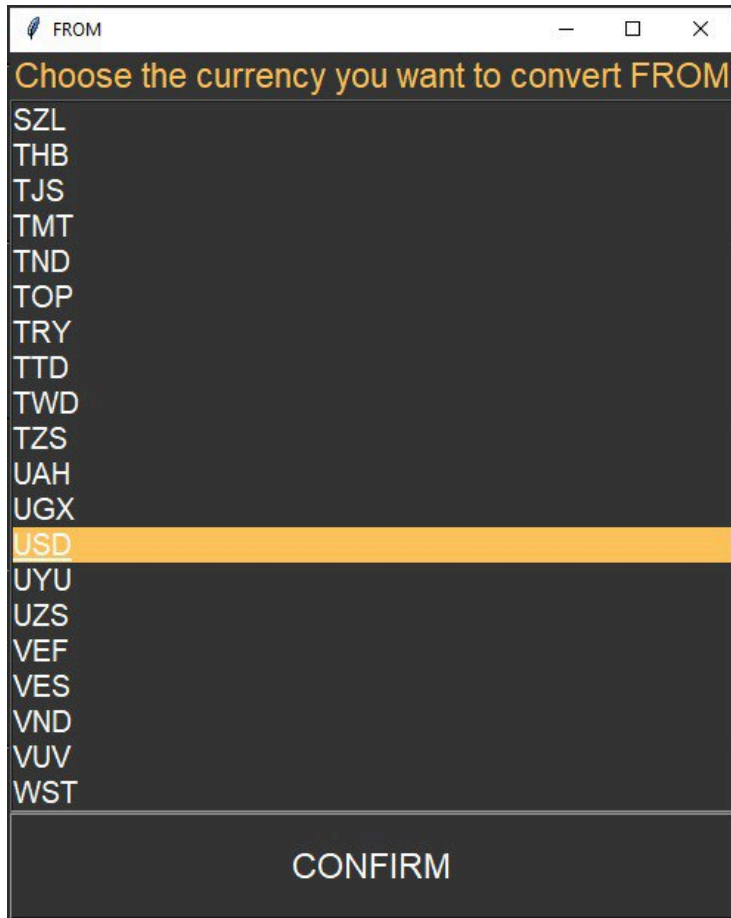
1. Запуск программы -> открывается
основное окно

The screenshot shows a web browser window titled "CurrencyConverter.de". The interface is dark-themed and consists of several stacked rectangular buttons. From top to bottom, the buttons are labeled: "FROM", "TO", "AMOUNT", "Waiting for conversion data..." (in orange text), "CONVERT", "CLEAR", and "FINISH".



Пример работы программы

2. Выбор валюты, из которой выполняется перевод. Выбор в отдельном всплывающем окне после нажатия кнопки FROM. Аналогично с целевой валютой после нажатия кнопки TO.





Пример работы программы

18

3. Ввод суммы для конвертирования (в отдельном окне, после нажатия AMOUNT).

A small window titled "AMOUNT" with a dark background. It contains a text input field with the value "1000.5" and a button labeled "CONFIRM". Above the input field, the text "Enter the amount you want to convert" is displayed in orange.

A larger window titled "CurrencyConverter.de" showing a sequence of screens. The screens are: "USD", "RUB", "1000.5", "Waiting for conversion data...", "CONVERT", "CLEAR", and "FINISH". The text is displayed in orange or white on a dark background.



Пример работы программы

4. Конвертируем с помощью кнопки CONVERT и получаем результат

The screenshot shows a web browser window titled "CurrencyConverter.de". The interface consists of a vertical stack of eight dark gray rectangular buttons. The first three buttons are labeled "USD", "RUB", and "1000.5" in orange text. The fourth button is labeled "Result: 80833.35 RUB" in orange text. The last three buttons are labeled "CONVERT", "CLEAR", and "FINISH" in white text.

5. Очищаем все введенные данные с помощью кнопки CLEAR. Можем завершить программу и закрыть окно с помощью кнопки FINISH

The screenshot shows the same web browser window titled "CurrencyConverter.de". The interface consists of a vertical stack of eight dark gray rectangular buttons. The first three buttons are labeled "FROM", "TO", and "AMOUNT" in white text. The fourth button is labeled "Waiting for conversion data..." in orange text. The last three buttons are labeled "CONVERT", "CLEAR", and "FINISH" in white text.



Другие примеры

20

1400 Армянских драм -> Евро

The screenshot shows a web browser window titled 'CurrencyConverter.de'. The interface has a dark background with orange text for currency codes and values. The first field contains 'AMD', the second 'EUR', and the third '1400.0'. Below these, the result is displayed as 'Result: 3.38 EUR'. At the bottom, there are four buttons: 'CONVERT', 'CLEAR', and 'FINISH' in white text, and an additional unlabeled button at the very bottom.

AMD
EUR
1400.0
Result: 3.38 EUR
CONVERT
CLEAR
FINISH

1 Bitcoin -> Доллары

The screenshot shows a web browser window titled 'CurrencyConverter.de'. The interface has a dark background with orange text for currency codes and values. The first field contains 'BTC', the second 'USD', and the third '1.0'. Below these, the result is displayed as 'Result: 26980.52 USD'. At the bottom, there are four buttons: 'CONVERT', 'CLEAR', and 'FINISH' in white text, and an additional unlabeled button at the very bottom.

BTC
USD
1.0
Result: 26980.52 USD
CONVERT
CLEAR
FINISH



Проверка некорректных данных

При попытке ввода отрицательного значения или перевода без выбора хотя бы одной валюты выдается ошибка.

The screenshot shows a web browser window with the title 'CurrencyConverter.de'. The interface consists of a vertical stack of seven dark gray buttons with white text. From top to bottom, the buttons are labeled: 'FROM', 'TO', 'AMOUNT', 'ERROR', 'CONVERT', 'CLEAR', and 'FINISH'. The 'ERROR' button is highlighted in a yellowish-orange color, indicating an active error state. The browser window also shows standard window control buttons (minimize, maximize, close) in the top right corner.



Спасибо за внимание!