profile_input

April 15, 2024

1 Profile Input

```python
[14]: #@title 1: Load libraries
      import ipywidgets as widgets
      import pandas as pd
      import os  # Make sure to include this import
      from IPython.display import display, HTML
```

```python
[8]: #@title 2: Creating text widgets for user input
     name = widgets.Text(description='Name:')
     address = widgets.Text(description='Address:')
     telephone_number = widgets.Text(description='Tel:')
     social_security_number_or_taxpayer_identification_number = widgets.
      ↪Text(description='ID (SS or EIN):')
     email = widgets.Text(description='E-Mail:')
     dob = widgets.DatePicker(description='D.O.B.:')  # Using DatePicker for date of␣
      ↪birth
     employment_status = widgets.FloatText(description='Employment Status (E=1,U=0):
      ↪')
     whether_you_are_employed_by_a_brokerage_firm = widgets.
      ↪IntText(description='Brokerage Firm (0=No, 1=Yes):')
     annual_income = widgets.IntText(description='Annual Income:')
     tax_status = widgets.Text(description='Tax Status␣
      ↪(M=Married,W=Widow,S=Single,D=Divorced):')
     trading_experience = widgets.IntText(description='Investment Experience (0-10):
      ↪')
     net_worth = widgets.IntText(description='Net Worth:')
     financial_knowledge = widgets.IntText(description='Financial Knowledge (0-10):')
```

```python
[9]: #@title 3: Button to trigger the input
     input_btn = widgets.Button(
         description='Save Data',
         button_style='success',
         tooltip='Click to save data',
         style={'button_color': 'lightgreen'}
     )
     # Output widget to display the input result
```

```python
output = widgets.Output()
```

[12]:
```python
#@title 4: Function to handle button click and save data to CSV
def on_button_clicked(b):
    with output:
        output.clear_output()
        # Collecting the data into a DataFrame
        data = pd.DataFrame([[
            name.value, address.value, email.value, dob.value.
 strftime('%Y-%m-%d') if dob.value else '',
            social_security_number_or_taxpayer_identification_number.value,
 employment_status.value,
            whether_you_are_employed_by_a_brokerage_firm.value, annual_income.
 value, tax_status.value,
            trading_experience.value, net_worth.value, financial_knowledge.value
        ]], columns=[
            'Name', 'Address', 'Email', 'DOB', 'SSN/TIN', 'Employment Status',
            'Employed by Brokerage Firm', 'Annual Income', 'Tax Status',
 'Trading Experience',
            'Net Worth', 'Financial Knowledge'
        ])
        # Check if the CSV file exists to decide whether to write headers
        file_exists = os.path.isfile('input_data.csv')

        # Saving data to CSV
        data.to_csv('input_data.csv', mode='a', header=not file_exists,
 index=False)
        print("Data saved successfully!")


input_btn.on_click(on_button_clicked)
```

[15]:
```python
#@title 5: Profile input form
message = "Please enter the required fields below for Profile setup"
display(HTML(f"<b><font size=4>{message}</font></b>"))

# Display the widgets
widgets.VBox([
    name, address, email, dob,
 social_security_number_or_taxpayer_identification_number, employment_status,
    whether_you_are_employed_by_a_brokerage_firm, annual_income, tax_status,
 trading_experience, net_worth,
    financial_knowledge, input_btn, output
])
```

```
<IPython.core.display.HTML object>
```

```
VBox(children=(Text(value='Test 123', description='Name:'), Text(value='34445␣
 ↪Six st. Los Angeles CA 91301', d…
```