

Adatbányászat a Gyakorlatban

6. Gyakorlat: Haladó Dash módszerek

Kuknyó Dániel
Budapesti Gazdasági Egyetem

2024/25
1.félév

1 Dinamikus felhasználói komponensek

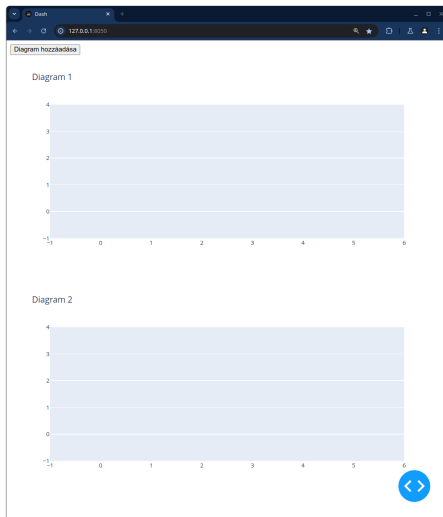
1 Dinamikus felhasználói komponensek

Dinamikus felhasználói komponensek (dyn_component_app_v1.py)

Olyan dinamikus komponensek, amelyek nem állandóak, hanem felhasználói interakcióra kerülnek hozzáadásra az alkalmazáshoz, és el is lehet őket távolítani.

Ehhez tartozóan az első alkalmazás elrendezése:

```
1 app.layout = html.Div([  
2     dbc.Button("Diagram hozzáadása", id='  
3         dyn_component_button'),  
4     html.Div(id='dyn_component_output',  
5         children=[]),  
6 ])
```



Callback függvény felhasználói komponensek hozzáadására

A callback a `dyn_component_output` Div children komponensét frissíti.

Ha a gombot megnyomták, létrehoz egy új oszlopdiagramot, és a diagram címében megjeleníti a gombnyomások számát. Az új diagramot hozzáadja a children listához.

Visszatér a frissített children listával, amely tartalmazza az új diagramot.

```
1 @app.callback(  
2     Output('dyn_component_output', '  
3         children'),  
4     Input('dyn_component_button', '  
5         n_clicks'),  
6     State('dyn_component_output', '  
7         children')  
8 )  
9 def add_new_chart(n_clicks, children):  
10     if not n_clicks:  
11         return no_update  
12     new_chart = dcc.Graph(figure=px.bar(  
13         title=f"Diagram {n_clicks}")  
14     )  
15     children.append(new_chart)  
16     return children
```

Az alkalmazás callback gráfja

A callback gráf ebben az esetben egy speciális hurkot mutat. Ez azt jelenti, hogy a függvény egy olyan komponenst térít vissza, amit megkapott paraméterül és módosított a függvénytörzsben.

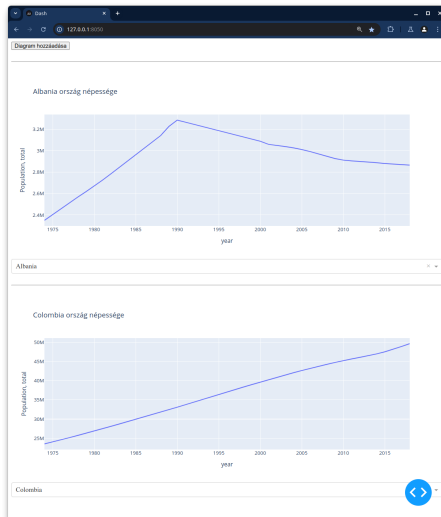
Ez a diagram változatlan marad a hozzáadott komponensek számától függetlenül.



Mintaillesztő callback függvények

Az alkalmazás következő verziója interaktivitást ad a felhasználói komponenseknek, mintaillesztő callback függvények segítségével.

Minden diagramnak van egy saját legördülő menüje, ahol ki lehet választani egy országot, majd kirajzolja a népességet a diagramra.



Dinamikus komponens elnevezések

Dash rendszerben egy komponens id paramétere bármilyen hash-képes (egy függvényen keresztül egyértelműen leképezhető) objektum lehet, így pl. egy szótár is.

Az alkalmazásban a id attribútum egy szótár, amely tartalmazza a type és index kulcsokat. Ez a struktúra lehetővé teszi komponenseket dinamikus azonosítását az alkalmazásban. Például több diagram létrehozása és módosítása esetén mindegyiket egyedi névvel látja el.

```
1 # Új diagram létrehozása
2 new_chart = dcc.Graph(
3     id={'type': 'chart', 'index':
4         n_clicks},
5     figure=px.bar(title=f"Diagram {
6         n_clicks}")
7 )
8 # Legördülő lista opciók létrehozása
9 countries = poverty[poverty['is_country']
10                      ][['Country Name']].drop_duplicates
11                      ().sort_values()
12 # Legördülő lista létrehozása
13 new_dropdown = dcc.Dropdown(
14     id={'type': 'dropdown', 'index':
15         n_clicks},
16     options=[{'label': c, 'value': c} for
17               c in countries],
18     placeholder='Ország kiválasztása'
19 )
```


Dash MATCH

A `dash.dependencies.MATCH` lehetővé teszi, hogy a callback egy adott komponenscsoport egy adott példányára vonatkozzon.

Például ha több dinamikus objektum közül mindegyiknek van egy egyedi `id` paramétere egy szótár formájában, akkor a `MATCH` segítségével meg lehet határozni, hogy a callback csak az adott indexnek megfelelő komponensre reagáljon.

```
1 @app.callback(  
2     Output({'type': 'chart', 'index':  
3         MATCH}, 'figure'),  
4     Input({'type': 'dropdown', 'index':  
5         MATCH}, 'value'),  
6 )  
7 def create_population_chart(country):  
8     if not country:  
9         return no_update  
10    # Adatkészlet szűrése  
11    df = poverty[poverty['Country Name']  
12        == country]  
13    # Diagram létrehozása  
14    fig = px.line(  
15        df,  
16        x='year',  
17        y='Population, total',  
18        title=f'{country} ország népessége'  
19    )  
20    return fig
```