Tutorial Lengkap Membuat IDP Program Dashboard (Dash + Python)

1. Tujuan

Tutorial ini membantu pengguna membuat dashboard **interaktif** untuk menganalisis data IDP (Individual Development Plan) dengan visualisasi:

- Pie chart kategori program (Leadership, Technical, Others)
- Bar chart jumlah peserta tiap program E1
- Tabel peserta kategori Others
- Bar chart per Development Objectives
- Top 10 Development Objectives

Dashboard ini dibuat menggunakan **Python** dan **Dash** (Plotly).

---

2. Persiapan Awal

2.1 Install Python

- Pastikan **Python 3.9+** sudah terinstall.
    - Bisa download di: https://www.python.org/downloads/
    - Saat instalasi, centang opsi **"Add Python to PATH"** agar bisa digunakan di terminal.

Cek versi Python di terminal / CMD:

python --version

Harus muncul versi Python seperti Python 3.11.5.

2.2 Install Library yang Dibutuhkan

Buka **Command Prompt** (Windows) atau **Terminal** (Mac/Linux), lalu jalankan perintah:

pip install pandas plotly dash openpyxl

Penjelasan library:

- pandas → untuk membaca dan mengolah data Excel
- plotly → untuk membuat grafik interaktif
- dash → untuk membuat dashboard web interaktif
- openpyxl → untuk membaca dan menulis file Excel .xlsx

2.3 Persiapkan File Excel

Dashboard membutuhkan **file Excel hasil klasifikasi program E1**:

- File harus bernama E1_fixx_program.xlsx (bisa sesuaikan, nanti ganti di script).
- Letakkan file di folder tertentu, misalnya:

D:\IDP FINAL\E1_fixx_program.xlsx

File Excel minimal memiliki kolom:

1. NAME → nama peserta
2. E1 → nama program E1
3. Kategori Program → Leadership / Technical / Others
4. Development Objectives → tujuan pengembangan peserta

Pastikan kolom **nama persis sama**.

---

3. Struktur Folder yang Disarankan

D:\IDP FINAL\

├— E1_fixx_program.xlsx # File data input

├— idp_dashboard.py # File script Python

---

4. Menulis Script Python Dashboard

Buat file baru **idp_dashboard.py** di folder yang sama, lalu masukkan seluruh codingan berikut (yang sudah kamu berikan):

```
# ------------------------ IMPORT LIBRARY -------------------------
```

```
import pandas as pd
from dash import Dash, html, dcc
import plotly.express as px

# ------------------------- LOAD DATA -------------------------
file_path = "D:\\IDP FINAL\\E1_fixx_program.xlsx"  # ganti sesuai lokasi file
df = pd.read_excel(file_path)

# ------------------------- HITUNG RINGKASAN -------------------------
total_peserta = df['NAME'].nunique()  # jumlah peserta unik

# ------------------------- KATEGORI PROGRAM -------------------------
df['Kategori Program'] = df['Kategori Program'].astype(str).str.strip()
leadership_count = df[df['Kategori Program']=='Leadership Program']['NAME'].nunique()
technical_count = df[df['Kategori Program']=='Technical Program']['NAME'].nunique()
others_count = df[df['Kategori Program']=='Others']['NAME'].nunique()

# ------------------------- PIE CHART -------------------------
pie_df = pd.DataFrame({
    'Program Type': ['Leadership Program', 'Technical Program', 'Others'],
    'Jumlah': [leadership_count, technical_count, others_count]
})

fig_pie = px.pie(
    pie_df,
```

```
    names='Program Type',
    values='Jumlah',
    title='Distribusi Kategori Program',
    color='Program Type',
    color_discrete_map={'Leadership Program':'#1f77b4',
                        'Technical Program':'#2ca02c',
                        'Others':'#7f7f7f'}
)
fig_pie.update_traces(textinfo='percent+value')
fig_pie.update_layout(title_x=0.5)

# ------------------------- MAP PROGRAM E1 KE LABEL SINGKAT -------------------------
program_map = {
    'Essential Professional Program (EPP)': 'EPP',
    'Supervisory Development Program (SDP)': 'SDP',
    'Management Development Program (MDP)': 'MDP',
    'People Manager 101': 'PM101',
    'Leader as Coach': 'LaC',
    'Young Professional Program': 'YPP',
    'Advanced Development Program (ADP)': 'ADP',
    'Project Management Excellence (PMX)': 'PMX',
    'Maintenance Inspector Program': 'MIP',
    'Preventive Maintenance Engineers (PME)': 'PME',
    'Cement Manufacturing Course': 'CMC',
    'Analyst Excellence (AX)': 'AX',
```

```
    'Basic Maintenance': 'BM',
    'Finance for non Finance': 'Finance',
    'Packer Excellence': 'Packer',
    'Patroller Excellence': 'Patroller',
    'Kiln Operator Excellence (KOX)': 'KOX',
    'Mill Operation Excellence (MOX)': 'MOX'
}

def categorize_e1_short(program_name):
    program_name = str(program_name).strip()
    if program_name.lower().startswith('training'):
        return 'Others'
    elif program_name in program_map:
        return program_map[program_name]
    else:
        return 'Others'

df['E1_short'] = df['E1'].apply(categorize_e1_short)

# ------------------------- KATEGORI PROGRAM OTOMATIS DARI E1 -------------------------
leadership_labels = ['EPP','SDP','MDP','PM101','LaC','YPP','ADP','Others']
technical_labels =
['PMX','MIP','PME','CMC','AX','BM','Finance','Packer','Patroller','KOX','MOX']

def auto_category(x):
    if x in leadership_labels:
```

```
        return 'Leadership Program'
    elif x in technical_labels:
        return 'Technical Program'
    else:
        return 'Others'

df['Kategori Program Auto'] = df['E1_short'].apply(auto_category)

# ------------------------- BAR CHART LEADERSHIP (tanpa Others) -------------------------
leadership_only_counts = df[df['E1_short'].isin([x for x in leadership_labels if x != 'Others'])]
leadership_only_counts = leadership_only_counts['E1_short'].value_counts().reindex([x for x in leadership_labels if x != 'Others'], fill_value=0).reset_index()
leadership_only_counts.columns = ['Program', 'Jumlah']

fig_leadership = px.bar(
    leadership_only_counts,
    x='Program',
    y='Jumlah',
    title='Jumlah Peserta Leadership Program',
    text='Jumlah',
    color='Program',
    color_discrete_sequence=px.colors.qualitative.Plotly
)
fig_leadership.update_traces(textposition='outside')

# ------------------------- BAR CHART TECHNICAL -------------------------
```

```
technical_counts = df[df['E1_short'].isin(technical_labels)]

technical_counts = technical_counts['E1_short'].value_counts().reindex(technical_labels, fill_value=0).reset_index()

technical_counts.columns = ['Program', 'Jumlah']


fig_technical = px.bar(
   technical_counts,
   x='Program',
   y='Jumlah',
   title='Jumlah Peserta Technical Program',
   text='Jumlah',
   color='Program',
   color_discrete_sequence=px.colors.qualitative.Set3
)
fig_technical.update_traces(textposition='outside')


# ------------------------ TABLE OTHERS ------------------------

others_counts = df[df['Kategori Program']=='Others']

others_table = others_counts.groupby('E1_short')['NAME'].nunique().reset_index()

others_table.columns = ['Program', 'Jumlah']


total_others = others_table['Jumlah'].sum()

total_row = pd.DataFrame({'Program':['Total Others'], 'Jumlah':[total_others]})

others_table = pd.concat([others_table, total_row], ignore_index=True)


html_others_table = html.Div([
```

```
    html.H4("Jumlah Peserta Others", style={'textAlign':'center','margin-bottom':'10px'}),
    html.Table([
        html.Thead(
            html.Tr([html.Th(col, style={'border':'1px solid black','padding':'5px','background-color':'#f2f2f2'}) for col in others_table.columns])
        ),
        html.Tbody([
            html.Tr([html.Td(others_table.iloc[i][col], style={'border':'1px solid black','padding':'5px'}) for col in others_table.columns])
            for i in range(len(others_table))
        ])
    ], style={'width':'40%','margin':'auto','border-collapse':'collapse','box-shadow':'2px 2px 8px #aaa','padding':'10px'})
])


# ------------------------- BAR CHART DEVELOPMENT OBJECTIVES -------------------------
bar_dev_obj = df['Development Objectives'].value_counts().reset_index()
bar_dev_obj.columns = ['Development Objectives', 'Jumlah']
fig_bar_dev = px.bar(
    bar_dev_obj,
    x='Development Objectives',
    y='Jumlah',
    title='Jumlah Peserta per Development Objectives',
    color='Jumlah',
    color_continuous_scale='Viridis',
    text='Jumlah'
```

```
)

fig_bar_dev.update_traces(textposition='outside')

fig_bar_dev.update_layout(xaxis_tickangle=-45)


top10_dev_obj = bar_dev_obj.head(10)


# ------------------------ DASH APP ------------------------

app = Dash(__name__)


app.layout = html.Div([

   html.H1("IDP Program Dashboard", style={'textAlign':'center','font-family':'Arial','margin-bottom':'30px'}),

   html.Div([dcc.Graph(figure=fig_pie)], style={'width':'50%','margin':'auto','margin-bottom':'40px'}),

   html.Div([

      html.Div(f"Total Peserta\n{total_peserta}", style={'padding':'20px','border-radius':'10px','background-color':'#f0f0f0','display':'inline-block','margin':'10px','text-align':'center','width':'180px','font-weight':'bold','box-shadow':'2px 2px 8px #aaa'}),

      html.Div(f"Leadership Program\n{leadership_count}", style={'padding':'20px','border-radius':'10px','background-color':'#1f77b4','color':'white','display':'inline-block','margin':'10px','text-align':'center','width':'180px','font-weight':'bold','box-shadow':'2px 2px 8px #aaa'}),

      html.Div(f"Technical Program\n{technical_count}", style={'padding':'20px','border-radius':'10px','background-color':'#2ca02c','color':'white','display':'inline-block','margin':'10px','text-align':'center','width':'180px','font-weight':'bold','box-shadow':'2px 2px 8px #aaa'}),

      html.Div(f"Others\n{others_count}", style={'padding':'20px','border-radius':'10px','background-color':'#7f7f7f','color':'white','display':'inline-block','margin':'10px','text-align':'center','width':'180px','font-weight':'bold','box-shadow':'2px 2px 8px #aaa'}),
```

```
    ], style={'textAlign':'center','margin-bottom':'50px'}),

    html.Div([

        html.Div([dcc.Graph(figure=fig_leadership)], style={'width':'48%', 'display':'inline-block'}),

        html.Div([dcc.Graph(figure=fig_technical)], style={'width':'48%', 'display':'inline-block',
'margin-left':'4%'}),

    ], style={'margin-bottom':'30px'}),

    html.Div([html_others_table], style={'margin-bottom':'50px'}),

    html.Div([dcc.Graph(figure=fig_bar_dev)], style={'width':'95%','margin':'auto','margin-
bottom':'50px'}),

    html.H3("Top 10 Development Objectives", style={'textAlign':'center','margin-
bottom':'20px'}),

    html.Div([

        html.Table([

            html.Thead(html.Tr([html.Th(col, style={'border':'1px solid
black','padding':'5px','background-color':'#f2f2f2'}) for col in top10_dev_obj.columns])),

            html.Tbody([html.Tr([html.Td(top10_dev_obj.iloc[i][col], style={'border':'1px solid
black','padding':'5px'}) for col in top10_dev_obj.columns]) for i in range(len(top10_dev_obj))])

        ], style={'width':'60%','margin':'auto','border-collapse':'collapse','box-shadow':'2px 2px
8px #aaa','padding':'10px'})

    ])

])


# ------------------------ RUN APP ------------------------

if __name__ == '__main__':

    app.run(debug=True)
```

5. Menjalankan Dashboard

1. Buka **Command Prompt** atau **Terminal**
2. Masuk ke folder script:

cd "D:\IDP FINAL"

3. Jalankan script:

python idp_dashboard.py

4. Tunggu beberapa detik, Dash akan memberi link seperti:

Running on http://127.0.0.1:8050/

5. Buka browser dan buka alamat tersebut → dashboard akan muncul.

---

6. Penjelasan Setiap Bagian

- **Import Library** → memanggil tools Python untuk Excel dan Dashboard
- **Load Data** → membaca file Excel E1_fixx_program.xlsx
- **Ringkasan Peserta** → menghitung total peserta, leadership, technical, others
- **Pie Chart** → distribusi peserta per kategori program
- **Map Program E1** → singkatan program dan kategori otomatis
- **Bar Chart Leadership & Technical** → jumlah peserta tiap program
- **Tabel Others** → menampilkan peserta yang programnya lain/unspecified
- **Bar Chart Development Objectives** → jumlah peserta tiap development objective
- **Top 10 Development Objectives** → menampilkan 10 terbesar
- **Dash Layout** → mengatur tampilan dashboard
- **Run App** → menjalankan dashboard di browser

---

7. Tips Tambahan

- Pastikan kolom Excel tidak ada spasi ekstra, nama kolom sesuai (NAME, E1, Kategori Program, Development Objectives)

- Tutup file Excel sebelum menjalankan script
- Gunakan browser Chrome/Edge untuk tampilan terbaik
- Jika ada error, cek kembali **path file** dan library yang terinstal