Diagrams

› **Guides**

Diagrams

`Diagram` is a primary object representing a diagram.

Basic

`Diagram` represents a global diagram context.

You can create a diagram context with the `Diagram` class. The first parameter of the `Diagram` constructor will be used to generate the output filename.

```python
from diagrams import Diagram
from diagrams.aws.compute import EC2

with Diagram("Simple Diagram"):
    EC2("web")
```

If you run the above script with the command below,

```
$ python diagram.py
```

it will generate an image file with single `EC2` node drawn as `simple_diagram.png` in your working directory and open that created image file immediately.

Jupyter Notebooks

Diagrams can also be rendered directly inside Jupyter notebooks like this:

```python
from diagrams import Diagram
from diagrams.aws.compute import EC2
```

```
with Diagram("Simple Diagram") as diag:
    EC2("web")
diag
```

Options

You can specify the output file format with the `outformat` parameter. The default is **png**.

> Allowed formats are: png, jpg, svg, pdf, and dot

```
from diagrams import Diagram
from diagrams.aws.compute import EC2

with Diagram("Simple Diagram", outformat="jpg"):
    EC2("web")
```

The `outformat` parameter also supports a list to output all the defined outputs in one call:

```
from diagrams import Diagram
from diagrams.aws.compute import EC2

with Diagram("Simple Diagram Multi Output", outformat=["jpg", "png", "dot"]):
    EC2("web")
```

You can specify the output filename with the `filename` parameter. The extension shouldn't be included, it's determined by the `outformat` parameter.

```
from diagrams import Diagram
from diagrams.aws.compute import EC2

with Diagram("Simple Diagram", filename="my_diagram"):
    EC2("web")
```

You can also disable the automatic file opening by setting the `show` parameter to **false**. The default is **true**.

```
from diagrams import Diagram
```

```python
from diagrams.aws.compute import EC2

with Diagram("Simple Diagram", show=False):
    EC2("web")
```

Diagrams also allow custom Graphviz dot attributes options.

> `graph_attr`, `node_attr` and `edge_attr` are supported. Here is a [reference link](#).

```python
from diagrams import Diagram
from diagrams.aws.compute import EC2

graph_attr = {
    "fontsize": "45",
    "bgcolor": "transparent"
}

with Diagram("Simple Diagram", show=False, graph_attr=graph_attr):
    EC2("web")
```

Last updated on 4/13/2024

Docs

Getting Started

Guides

Nodes

More

Star 40,612