Line-by-Line Explanation of the Initial Code

The file waveguide\_pcell.py defines a Python script for creating a simple Parametric Cell (PCell)  for a waveguide in KLayout, a popular layout editor for photonics and microelectronics. The script utilizes SiEPIC-Tools, which is a library for photonic circuit design.

Here is an explanation of the key components in the file:

**1. Imports and Utilities**

5| import pya

6| import SiEPIC

7| from SiEPIC.utils import get\_technology\_by\_name

* **pya**: This is the Python API for KLayout, providing the framework for developing custom cells.
* **SiEPIC**: A library used for photonic circuit design.
* **get\_technology\_by\_name**: A utility function to fetch information about a specific technology  (e.g., "EBeam").

### 2. ****WaveguidePCell Class****

9| class WaveguidePCell(pya.PCellDeclarationHelper):

This class defines the PCell for the waveguide. It inherits from PCellDeclarationHelper, which simplifies defining parametric cells.

#### ****Constructor (****\_\_init\_\_****)****

10| def \_\_init\_\_(self):

11| super().\_\_init\_\_()

12| self.param("width", self.TypeDouble, "Waveguide Width (um)", default=0.5)

13| self.param("length", self.TypeDouble, "Waveguide Length (um)", default=10.0)

* Parameters of the waveguide:
  + **width**: The width of the waveguide, defaulting to 0.5 µm.
  + **length**: The length of the waveguide, defaulting to 10 µm.

#### ****Display Text (****display\_text\_impl****)****

15| def display\_text\_impl(self):

16| return f"Waveguide (width={self.width}, length={self.length})"

* Defines the text that will be displayed for the PCell in the KLayout GUI.

#### ****Produce Implementation (****produce\_impl****)****

18| def produce\_impl(self):

19| tech = get\_technology\_by\_name("EBeam")

20| dbu = self.layout.dbu

21| width = self.width

22| length = self.length

23| points = [

24| pya.DPoint(0, -width/2),

25| pya.DPoint(length, -width/2),

26| pya.DPoint(length, width/2),

27| pya.DPoint(0, width/2)

28| ]

29| shape = pya.DPolygon(points)

30| self.cell.shapes(self.layout.layer(tech["Si"])).insert(shape)

* **Technology**: Retrieves the "EBeam" technology using get\_technology\_by\_name.
* **Waveguide Shape**: Uses the width and length parameters to create a rectangular polygon representing the waveguide.
* **Placement**: Adds this polygon to the silicon layer (tech["Si"]) of the layout.

### 3. ****WaveguideLibrary Class****

33| class WaveguideLibrary(pya.Library):

34| def \_\_init\_\_(self):

35| self.description = "Waveguide PCell Library"

36| self.layout().register\_pcell("Waveguide", WaveguidePCell())

37| self.register("WaveguideLibrary")

* Registers the WaveguidePCell in a custom library called WaveguideLibrary.
* This library will appear in KLayout, allowing users to instantiate the waveguide PCell.

### 4. ****Library Registration****

39| WaveguideLibrary()

* Instantiates and registers the WaveguideLibrary to make it accessible in KLayout.

### Summary

This script defines a parametric waveguide cell that users can customize by specifying its width and length. It integrates seamlessly with KLayout and SiEPIC-Tools, making it a useful tool for photonic circuit designers.