```
!pip install gradio
```

Collecting semantic-version~=2.0 (from gradio)
  Downloading semantic_version-2.10.0-py2.py3-none-any.whl.metadata (9.7 kB)
Collecting starlette<1.0,>=0.40.0 (from gradio)
  Downloading starlette-0.46.2-py3-none-any.whl.metadata (6.2 kB)
Collecting tomlkit<0.14.0,>=0.12.0 (from gradio)
  Downloading tomlkit-0.13.2-py3-none-any.whl.metadata (2.7 kB)
Requirement already satisfied: typer<1.0,>=0.12 in /usr/local/lib/python3.11/dis
Requirement already satisfied: typing-extensions~=4.0 in /usr/local/lib/python3.
Collecting uvicorn>=0.14.0 (from gradio)
  Downloading uvicorn-0.34.2-py3-none-any.whl.metadata (6.5 kB)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages
Requirement already satisfied: websockets<16.0,>=10.0 in /usr/local/lib/python3.
Requirement already satisfied: idna>=2.8 in /usr/local/lib/python3.11/dist-packa
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.11/dist-pa
Requirement already satisfied: certifi in /usr/local/lib/python3.11/dist-package
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.11/dist-p
Requirement already satisfied: h11>=0.16 in /usr/local/lib/python3.11/dist-packa
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packag
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packag
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.11/dist-pa

```python
import gradio as gr

# Define separate functions for each operation
def add(a, b):
    return a + b

def subtract(a, b):
    return a - b

def multiply(a, b):
    return a * b

def divide(a, b):
    if b == 0:
        return "Error: Cannot divide by zero"
    return a / b

# Function to call the correct operation
def calculator(a, b, operation):
    if operation == "Addition":
        return add(a, b)
    elif operation == "Subtraction":
        return subtract(a, b)
    elif operation == "Multiplication":
        return multiply(a, b)
    elif operation == "Division":
        return divide(a, b)

# Create the Gradio interface
interface = gr.Interface(
    fn=calculator,
    inputs=[
        gr.Number(label="First Number"),
        gr.Number(label="Second Number"),
        gr.Radio(["Addition", "Subtraction", "Multiplication", "Division"], label="Choose O
    ],
    outputs=gr.Textbox(label="Result"),
    title="Simple Calculator",
    description="Enter two numbers and select an operation to perform calculations."
)

# Launch the Gradio app
interface.launch()
```

Simple Calculator

Enter two numbers and select an operation to perform calculations.

First Number

0

Second Number

0

Choose Operation

○ Addition   ○ Subtraction   ○ Multiplication   ○ Division

Clear   Submit

Result