Library Imports

In [ ]:
```
import numpy as np
import plotly.graph_objects as go
from scipy.integrate import quad
```

Method 1

In [ ]:
```
# Define the function to be integrated
def f(x):
    return 1 / (2 - np.sqrt(x))

# Integrate the function from 0 to 5
integral_value, _ = quad(f, 0, 5)

# Generate x values for plotting
x = np.linspace(0, 5, 1000)
y = f(x)
```

```
C:\Users\Aaron\AppData\Local\Temp\ipykernel_19532\3069320982.py:6: IntegrationWarnin
g: The integral is probably divergent, or slowly convergent.
  integral_value, _ = quad(f, 0, 5)
```

what my computer really means....

good thing Gaussian Quadrature is robust

$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} w_i \cdot f(x_i)$

In [ ]:
```
# Create the plotly figure
fig = go.Figure()

# Plot the function
fig.add_trace(go.Scatter(x=x, y=y, fill='tozeroy', mode='lines', name='y = 1/(2-√x)

# Add title and labels
fig.update_layout(title='Graph of y = 1/(2-√x)',
                  xaxis_title='x',
                  yaxis_title='y')

# Show the figure
fig.show()
```

In [ ]:
```
# Print the value of the integral
print(f"Value of I from 0 to 5: {integral_value:.5f}")
```

```
Value of I from 0 to 5: 4.07499
```

Method 2

In [ ]:
```
# Define the function to be integrated
def f(x):
    return 1 / (2 - np.sqrt(x))

# Integrate the function from 0 to 3.99 and from 4.01 to 5
integral_value_1, _ = quad(f, 0, 3.99999)
integral_value_2, _ = quad(f, 4.00001, 5)

# Sum the two integrals
total_integral = integral_value_1 + integral_value_2

print(f"Value of I from 0 to 5: {total_integral:.5f}")
```

```
Value of I from 0 to 5: 4.07500
```