Q1. Flask is a lightweight and extensible web framework for Python. It is designed to be simple and easy to use, making it an excellent choice for building web applications and APIs. Some advantages of Flask include:

Lightweight: Flask has minimal dependencies and a simple core, making it lightweight and easy to learn.

Flexibility: Flask allows developers to choose the components they need, enabling flexibility and customization.

Extensibility: Flask provides a modular architecture with support for extensions, allowing developers to add functionality as needed.

Built-in Development Server: Flask comes with a built-in development server, making it easy to get started with web development.

Jinja2 Templating: Flask uses the Jinja2 templating engine, which provides powerful and flexible template inheritance and dynamic content generation.

RESTful Request Handling: Flask supports RESTful request handling, making it easy to build APIs and web services.

Active Community: Flask has a large and active community of developers, providing plenty of resources, tutorials, and support.

Q2. Here's a simple Flask application to display "Hello World!!":

```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello World!!'

if __name__ == '__main__':
    app.run(debug=True)
```

To run this Flask application, save the code in a file named app.py and execute it using python app.py command in the terminal. You will see the "Hello World!!" message displayed in the browser.

Q3.

App routing in Flask refers to the process of mapping URL paths to view functions. We use app routes to define the behavior of the application when a specific URL is requested. App routes allow us to create different endpoints for handling various HTTP requests and serving different content to users.

Q4. Here's a Flask application with "/welcome" and "/" routes:

```
from flask import Flask

app = Flask(__name__)

@app.route('/welcome')
def welcome():
    return 'Welcome to ABC Corporation'

@app.route('/')
def company_details():
    return '''Company Name: ABC Corporation
    Location: India
    Contact Detail: 999-999-9999'''

if __name__ == '__main__':
    app.run(debug=True)
```

To access the "/welcome" route, go to http://localhost:5000/welcome in your browser, and to access the "/" route, go to http://localhost:5000/.

Q5. The url_for() function in Flask is used for URL building. It generates URLs for Flask view functions based on their names. Here's a Python code to demonstrate the working of the url_for() function:

```
from flask import Flask, url_for

app = Flask(__name__)

@app.route('/')
def index():
    return 'Index Page'
```

```
@app.route('/login')
def login():
    return 'Login Page'

@app.route('/user/<username>')
def profile(username):
    return f'User {username}'

with app.test_request_context():
    print(url_for('index'))
    print(url_for('login'))
    print(url_for('profile', username='john'))
```

This code will print the URLs for the index, login, and profile routes defined in the Flask application.