

# AeroAspire - SDE Intern

## Gokul Krishna S

Week 3 – Day 2 (October 8)

### Questions/Reflections:

1. What is difference between path parameters vs query parameters?

#### Path Parameters vs Query Parameters :

- **Path parameters** are part of the URL path and uniquely identify a specific resource (e.g., /users/123). They are required and define which resource the operation is acting on.
- **Query parameters** are appended after the ? in a URL and are typically used for filtering, sorting, or pagination (e.g., /users?city=Karachi). They are optional and used for modifying or filtering the operation on a resource or resources.

Example:

- /products/42 (Path parameter: identifies product ID 42)
- /products?category=books&sort=price (Query parameters: filters by category, sorts by price).

2. How to handle errors if client passes invalid id / resource not found.

- If the client passes an invalid ID (e.g., wrong type or format), return 400 Bad Request.
- If the client passes a well-formatted ID that does not correspond to any existing resource, return 404 Not Found.
- Example error response:

```
json{
  "error": "Resource not found."
}
```

- This enables clear distinction between input errors and non-existent resources for better client experience

3. Describe flow of updating a resource: client sends JSON → Flask handler → modify data → return response.

- Client sends a PUT/PATCH request with JSON to an endpoint like /items/123.
- Flask route handler extracts the path parameter (e.g., id=123) and parses the JSON payload using request.get\_json().
- Validate input and check if the resource exists. If not, return 404.
- Modify the resource in the database or data store.
- Return a response, usually the updated resource (with 200 OK) or a status/no content (204 No Content) on success.

- Example:

**Python**

```
```
```

```
@app.route('/items/<int:item_id>', methods=['PUT'])
def update_item(item_id):
    data = request.get_json()
    item = Item.query.get(item_id)
    if not item:
        return jsonify({'error': 'Item not found'}), 404
    # update fields from data here
    db.session.commit()
    return jsonify(item.to_dict()), 200
```

```
```
```