

# TEKNOLOGI MIKRO SERVICE

## LAPORAN HASIL PRAKTIKUM

### HASIL PRAKTIKUM

#### Praktikum 3

Nama : Michael Mervin Ruswan

NIM : 202310016

##### 1. Tabel Program Study

The screenshot displays a web application interface with a table titled "Insert Program Study". The table contains the following data:

id	code	department_id	description	faculty_id	is_active	name	program
1	T12	8	TI	2	1	Teknologi Informasi	1

Below the table, there is a section for "Query results operations" with buttons for Print, Copy to clipboard, Export, Display chart, and Create view. A "Bookmark this SQL query" section is also present with a label input field and a checkbox for "Let every user access this bookmark".

On the right side, there is an API client interface showing a collection of endpoints under "APLUBIK". The "Program Study" endpoint is selected, showing a POST request to "localhost:8080/api/programstudy". The request body is a JSON object:

```
{  "name": "Teknologi Informasi",  "description": "TI",  "code": "T12",  "program_id": 1,  "faculty_id": 2,  "department_id": 8,  "is_active": 1}
```

The response section shows a placeholder for the response data.

The screenshot displays a web application interface with a database table and an API client. The database table, named 'programstudy', contains the following data:

id	code	department_id	description	faculty_id	is_active	name	program_id
1	T12	8	TI	2	1	Teknologi Informasi	1
2	S12	9	SI	2	1	Sistem Informasi	2

The API client interface shows a POST request to the endpoint 'localhost:8080/api/programstudy'. The request body is a JSON object:

```

{
  "name": "Sistem Informasi",
  "description": "SI",
  "code": "S12",
  "program_id": 2,
  "faculty_id": 2,
  "department_id": 9,
  "is_active": 1
}

```

## Find All Program Study

The screenshot displays a web application interface with a database table and an API client. The database table, named 'programstudy', contains the following data:

id	code	department_id	description	faculty_id	is_active	name	program_id
1	T12	8	TI	2	1	Teknologi Informasi	1
2	S12	9	SI	2	1	Sistem Informasi	2

The API client interface shows a GET request to the endpoint 'localhost:8080/api/programstudy'. The response body is a JSON array of objects:

```

[
  {
    "id": 1,
    "name": "Teknologi Informasi",
    "description": "TI",
    "code": "T12",
    "program_id": 1,
    "faculty_id": 2,
    "department_id": 8,
    "is_active": true
  },
  {
    "id": 2,
    "name": "Sistem Informasi",
    "description": "SI",
    "code": "S12",
    "program_id": 2,
    "faculty_id": 2,
    "department_id": 9,
    "is_active": true
  }
]

```

## Update Program Study

The screenshot displays a web application interface on the left and an API client interface on the right. The web application shows a table with columns: id, code, department\_id, description, faculty\_id, is\_active, name, and program\_id. The table contains two rows of data. The API client interface shows a collection of endpoints under 'My Workspace' and 'API\_BIK'. The selected endpoint is 'POST Update Program Study' with a body containing JSON data.

id	code	department_id	description	faculty_id	is_active	name	program_id
1	T13	8	IT	2	1	Teknologi Informasi	1
2	S12	9	SI	2	1	Sistem Informasi	2

```
{
  "id": 1,
  "name": "Teknologi Informasi",
  "description": "IT",
  "code": "T13",
  "program_id": 1,
  "faculty_id": 2,
  "department_id": 8,
  "is_active": true
}
```

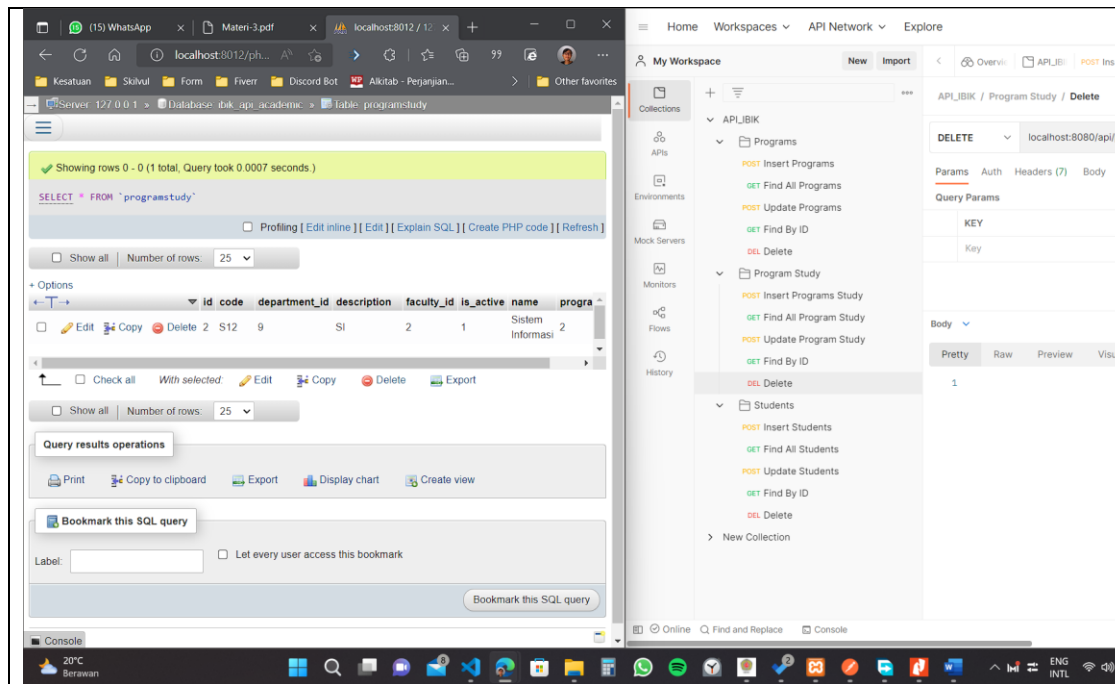
## Find By ID

The screenshot displays a web application interface on the left and an API client interface on the right. The web application shows a table with columns: id, code, department\_id, description, faculty\_id, is\_active, name, and program\_id. The table contains two rows of data. The API client interface shows a collection of endpoints under 'My Workspace' and 'API\_BIK'. The selected endpoint is 'GET Find By ID' with a body containing JSON data.

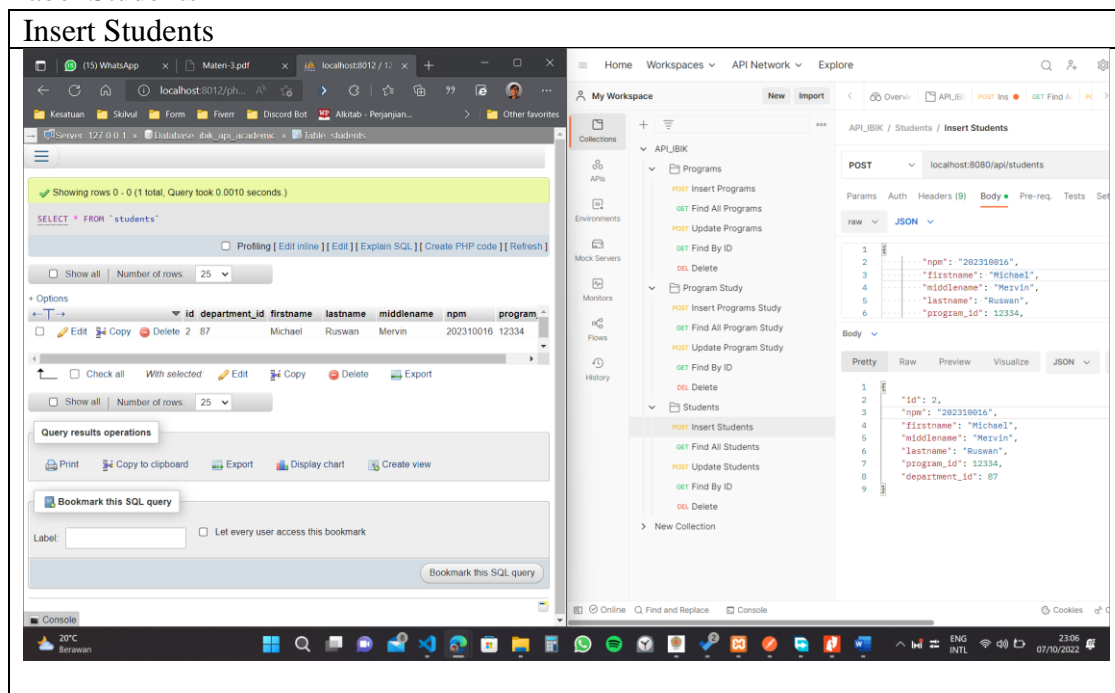
id	code	department_id	description	faculty_id	is_active	name	program_id
1	T13	8	IT	2	1	Teknologi Informasi	1
2	S12	9	SI	2	1	Sistem Informasi	2

```
{
  "id": 2,
  "name": "Sistem Informasi",
  "description": "SI",
  "code": "S12",
  "program_id": 2,
  "faculty_id": 2,
  "department_id": 9,
  "is_active": true
}
```

## Delete



## 2. Tabel Students



Screenshot of a web browser showing a REST client interface (Postman) and a database table view. The browser tabs include WhatsApp, Materi-3.pdf, and localhost:8012. The REST client shows a collection named 'API/BIK' with endpoints for 'Programs', 'Program Study', and 'Students'. The 'Students' endpoint is selected, showing a POST request to 'localhost:8080/api/students' with a JSON body:

```
{  "id": 3,  "npm": "202310117",  "firstname": "Laxrene",  "middlename": "Pol",  "lastname": "Euclair",  "program_id": 1234}
```

The database table view shows the 'students' table with columns: id, department\_id, firstname, lastname, middlename, npm, and program. The table contains 2 rows:

id	department_id	firstname	lastname	middlename	npm	program
2	87	Michael	Ruswan	Mervin	202310016	12334
3	87	Laxrene	Euclair	Pol	202310117	12334

## Find All Students

Screenshot of the REST client interface showing the 'Find All Students' endpoint. The endpoint is a GET request to 'localhost:8080/api/students'. The response body is shown in the 'Body' tab, displaying a JSON array of student objects:

```
[  {  "id": 2,  "npm": "202310016",  "firstname": "Mic",  "middlename": "Me",  "lastname": "Ruswan",  "program_id": 12334,  "department_id": 87  },  {  "id": 3,  "npm": "202310117",  "firstname": "Lax",  "middlename": "Po",  "lastname": "Euclair",  "program_id": 12334,  "department_id": 87  }]
```

## Update Students

Screenshot of a web browser showing a REST client interface (Postman) and a database table view. The database table view displays the following data:

id	department_id	firstname	lastname	middleinitial	npm	program
2	87	Michael	Ruswan	Mervin	202310016	12334
3	87	Marposia	Euclair	Pols	202310117	12334

The REST client interface shows a collection named 'API\_IBIK' with a sub-collection 'Students'. The 'Find By ID' endpoint is selected, and the response body is displayed in JSON format:

```
{  "id": 3,  "npm": "202310117",  "firstname": "Marposia",  "middleinitial": "Pols",  "lastname": "Euclair",  "program_id": 12334,  "department_id": 87}
```

## Find By ID

Screenshot of a web browser showing a REST client interface (Postman) and a database table view. The database table view displays the following data:

id	department_id	firstname	lastname	middleinitial	npm	program
2	87	Michael	Ruswan	Mervin	202310016	12334
3	87	Marposia	Euclair	Pols	202310117	12334

The REST client interface shows a collection named 'API\_IBIK' with a sub-collection 'Students'. The 'Find By ID' endpoint is selected, and the response body is displayed in JSON format:

```
{  "id": 2,  "npm": "202310016",  "firstname": "Michael",  "middleinitial": "Mervin",  "lastname": "Ruswan",  "program_id": 12334,  "department_id": 87}
```

## Delete

localhost:8012 / 1/2

Database: itak\_api\_academic > table: students

Showing rows 0 - 0 (1 total, Query took 0.0011 seconds.)

```
SELECT * FROM `students`
```

Profiling | Edit inline | Edit | Explain SQL | Create PHP code | Refresh

Show all | Number of rows: 25

id	department_id	firstname	lastname	middlename	npm	program
3	87	Marposia	Euclair	Pols	202310117	12334

Options

Check all | With selected | Edit | Copy | Delete | Export

Show all | Number of rows: 25

Query results operations

Print | Copy to clipboard | Export | Display chart | Create view

Bookmark this SQL query

Label:  ☐ Let every user access this bookmark

Bookmark this SQL query

Console

20°C Berawan

Home Workspaces API Network Explore

My Workspace

Collections

- APIs
  - Programs
    - POST Insert Programs
    - GET Find All Programs
    - POST Update Programs
    - GET Find By ID
    - DELETE Delete
  - Program Study
    - POST Insert Programs Study
    - GET Find All Program Study
    - POST Update Program Study
    - GET Find By ID
    - DELETE Delete
  - Students
    - POST Insert Students
    - GET Find All Students
    - POST Update Students
    - GET Find By ID
    - DELETE Delete
- New Collection

API/BIK / Students / Delete

DELETE localhost:8080/api/...

Params Auth Headers (7) Body

Query Params

KEY
Key

Body

Pretty Raw Preview Vis

1

Online Find and Replace Console