**1st Group Work. (5min)  
  
What could be data for PRODUCT in coffee house?**

**2nd Group Work (6min)**

**Auth API example**

|  |  |  |
| --- | --- | --- |
| **@desc** | **@route** | **@access** |
| Create New User | POST /api/v1/auth/register | Public |
| Login User | POST /api/v1/auth/login | Public |
| Logout User | GET /api/v1/auth/logout | Private |

**User API example**

|  |  |  |
| --- | --- | --- |
| **@desc** | **@route** | **@access** |
| Get All Users | GET /api/v1/users | Private/Admin |
| Get Single User | GET /api/v1/users/:userId | Private |
| Update User | PUT /api/v1/users/:userId | Private |
| Update Users Password | PATCH /api/v1/users/:userId | Private |
| Delete A User | DELETE /api/v1/users/:userId | Private/Admin |

**Make similar table as User for PRODUCT API!**

**3rd Group Work (7min)**

Select two routes of Product API.

Think of cases when we would need to refuse (send Error) when someone is using these routes?

How would you respond to client if everything is OK? (Please think of why – every time we send data without reason it is waste of resources)

* Success Message
* Data
* Old Data
* Updated Data
* Deleted Data
* Combination of message and data