Mutation with Variable

User Table
Add User

```
mutation createUser(
    $email: String!
    $password: String!
    $username: String!
    $empId: String!
    $department: String!
    $role: String!
    $designation: String!
    $teamleader: String!){
    createUser(input:{
      email: $email
      password: $password
      username: $username
      empid: $empId
      department: $department
      role: $role
      designation: $designation
      teamleader: $teamleader
    }){
      users{
        id
        email
      }
    }
  }
```

Login

```
mutation Login($email:String! $password:String! ){
 tokenAuth(email: $email, password: $password){
   success
   errors
   token
   refreshToken
  user{
      id
      email
      username
      firstName
      lastName
      teamleader
      empid
      role
      department
      designation
      address
      login
      logout
      isActive
   }
 }
}
```

Edit User

```
mutation updateuser(
    $id:String!
    $email: String!
    $username: String!
    $empId: String!
    $department: String!
    $role: String!
    $designation: String!
    $teamleader: String!)
{
  updateUser(input:{
      id: $id
      email: $email
      username: $username
      empid: $empId
      department: $department
      role: $role
      designation: $designation
      teamleader: $teamleader
  }){
    users{
      id
    }
  }
}
```

Logout

```
mutation Logout($id:String!){
  logout(id:$id){
    set
  }
}
```

Check In

```
mutation checkin($id:String!){
  checkInUpdate(id:$id){
    checkin{
      id
      user{
        id
      }
    }

  }
}
```

Check Out

```
mutation checkOut($id:String!){
  checkOut(id:$id){
    checkout{
      id
      user{
        id
      }
    }
```

```
  }
}
```

User Regularization Request

```
mutation regularizationRequest($userId:String!
  $checkin:String!
  $checkout:String!
  $reason:String!
  $teamleader:String!){
    addRegularization(input:{
      userId:$userId,
      checkIn:$checkin,
      checkOut:$checkout
      reason:$reason
      teamleaderId:$teamleader
    }){
      regularizationApproval{
        id
      }
    }
  }
```

Regularization Manager approval

```
mutation regularizationMangerApprove($id:String $status:String!){
  regularizationApprovalManager(id:$id status:$status){
    RegularizationApprovalManager{
      checkIn
      checkOut
      reason
      teamleaderId
      status
      user{
        id
      }
    }
  }
}
```

Regularization HR approval

```
mutation regularizationHRApprove($id:String $status:String!){
  regularizationApproval(id:$id status:$status){
    regularizationApprovalHR{
      checkIn
      checkOut
      reason
      teamleaderId
      status
      user{
        id
      }
    }
  }
}
```

Request For Leave

```
mutation requestForLeave($userId:String!
  $teamLeaderId:String!
  $requestedOn:Date!
  $leaveType:String!
  $leaveFrom:String!
```

```
    $leaveTill:String!
    $halfDay:String!
    $reason:String!
){
  addleaveApproval(input:{
    userId:$userId,
    teamleaderId:$teamLeaderId,
    requestedOn:$requestedOn,
    leaveType:$leaveType,
    leaveFrom:$leaveFrom,
    leaveTill:$leaveTill,
    halfDay:$halfDay,
    reason:$reason
  }){
    leaveApproval{
      teamleaderId
      requestedOn
      leaveType
      leaveFrom
      leaveTill
      totalDays
      halfDay
      reason
    }
  }
}
```

Leave approval manager

```
    mutation leaveManagerApproval($id:String! $status:String!){
  leaveapprovalmanager(id:$id status:$status){
    leaveApprovalManager{
      id
      requestedOn
      leaveType
      leaveFrom
      leaveTill
      totalDays
      halfDay
      reason
      status
      teamleaderId
      teamApproval
    }
  }
}
```

Leave approval hr

Pending

Add Manager

```
mutation addManager($id:String! $status:Boolean!){
  addmanagerApproval(id:$id status:$status){
    managerapproval{
      id
      status
      lastupdate
```

```
    user{
      id
      email
      username
    }
   }
  }
}
```

Add Team

```
     mutation addTeams($userId:String $managerId:String! $status:Boolean!){
  addTeams(input:{
   userId:$userId,
   managerId:$managerId,
   status:$status
  }){
   Team{
    user{
      id
      email
      empid
    }
    manager{
      id
    }
   }
  }
}
```

Delete User

```
 mutation deleteUser($id:String!){
  deleteUsers(id:$id){
   deleteusers
  }
}
```

Delete manager

```
mutation deleteManager($id:String!){
  managerdelete(id:$id){
   deletemanager
  }
}
```

Edit manager

```
mutation updateManager($id:String!){
  editManager(id:$id){
   managerupdate{
     id
     status
   }
  }
}
```

Add Leave

```
mutation addleave($id:String!
                   $casualLeave:String!
                   $sickLeave:String!
                   $earnedLeave:String!
                   $restrictedLeave:String!){
  addLeave(input:{
    userId:$id
    casualLeave:$casualLeave
    sickLeave:$sickLeave
    earnedLeave:$earnedLeave
    restrictedLeave:$restrictedLeave
  }){
    leaveConfiguration{
      id
      casualLeave
      sickLeave
      earnedLeave
      restrictedLeave
      user{
        id
        username
      }
    }
  }
}
```

Work from home request

```
mutation requestforWFH($id:String!
$startDate:String!
$endDate:String!
$teamleader_id:String
$reason:String!
){
  requestForWfh(input:{
    userId:$id
    startDate:$startDate
    endDate:$endDate
    teamleaderId:$teamleader_id
    reason:$reason
  }){
    wfhrequest{
      requestedDate
      startDate
      endDate
      reason
      totalDays
      workFromHome
      task
      teamApproval
      hrApproval
      lastUpdate
      user{
        id
        username
      }
    }
```

```
  }
}
```

Wfh Manager approval

```
mutation WFHManagerApproval($id:String!
  $status:String!
  $task:String!
){
  WfhManagerApproval(id:$id status:$status task:$task){
    wfhApprovalManager{
      requestedDate
      startDate
      endDate
      reason
      totalDays
      workFromHome
      task
      teamApproval
      hrApproval
      lastUpdate
      user{
        id
        username
      }
    }
  }
}
```

Wfh hr approval

```
mutation WFHHrApproval($id:String!
  $status:String!
){
  WfhApprovalHR(id:$id status:$status){
    wfhApprovalHR{
      requestedDate
      startDate
      endDate
      reason
      totalDays
      workFromHome
      task
      teamApproval
      hrApproval
      lastUpdate
      user{
        id
        username
      }
    }
  }
}
```