

Required time to perform the practical task:

This practical task is for 2 hours. You may stretch up to 2:30 hours if required for the quality task submission.

Please stop when you reach 2 hours if you have not completed task 1.

If you have time after task 1, you may begin task 2.

If you think you need more time than 2:30 hours, You can request in reply to the mail and we always encourage you to take more time to complete the task at your convenient time as effectively and efficiently as possible 😊

Note:

- Before starting the task you need to create a Repo with First Commit.
- After completion of the task, you need to Push your latest source on Git, make it Public and **share the Public repo link**. (Time will be counted between the First and Last commit)

Using Golang Gin and a MySQL database:

- It is preferred that you do not use an ORM. We like to see your use of MySQL (SQL) language to accomplish the task.
- It is much preferred that you complete this task in a single package (folder). You may use separate files for what would be separate packages if you desire.
- Please do not add any crud features or other endpoints to the code.
- Assume that a database has been created for you with the following table specifications and data:

```
create database cetec;
```

```
use cetec;
```

Person

```
create table person(id int not null key auto_increment, name varchar(255), age int);  
insert into person(id, name, age) values (1, "mike", 31), (2, "John", 20), (3, "Joseph", 20);
```

You can assume the following data in person:

```
Name, age, id
```

```
mike , 31, 1
```

```
John, 45, 2
```

```
Joseph, 20, 3
```

Phone

```
create table phone(id int not null key auto_increment, number varchar(255), person_id int);
insert into phone(id, person_id, number) values (1,1, "444-444-4444"), (8,2, "123-444-7777"),
(3,3, "445-222-1234");
```

You can assume the following data in phone:

person_id	id	number
-----------	----	--------

1	1	444-444-4444
---	---	--------------

2	8	123-444-7777
---	---	--------------

3	3	445-222-1234
---	---	--------------

Address

```
create table address(id int not null key auto_increment, city varchar(255), state varchar(255),
street1 varchar(255), street2 varchar(255), zip_code varchar(255));
insert into address(id, city, state, street1, street2, zip_code) values (1,"Eugene", "OR", "111
Main St", "", "98765"), (2, "Sacramento", "CA", "432 First St", "Apt 1", "22221"), (3, "Austin",
"TX", "213 South 1st St", "", "78704");
```

You can assume the following data in address:

id	city	state	street1	street2	zip_code
----	------	-------	---------	---------	----------

1	Eugene	OR	"111 Main St"	"	98765
---	--------	----	---------------	---	-------

2	Sacramento	CA	"432 First St"	"Apt 1"	22221
---	------------	----	----------------	---------	-------

3	Austin	TX	"213 South 1st St"	"	78704
---	--------	----	--------------------	---	-------

Address_join

```
create table address_join(id int not null key auto_increment, person_id int, address_id int);
insert into address_join(id, person_id, address_id) values (1,1,3),(2,2,1),(3,3,2);
```

You can assume the following data in address_join:

id	person_id	address_id
----	-----------	------------

1	1	3
---	---	---

2	2	1
---	---	---

3	3	2
---	---	---

Task 1:

make a REST endpoint (GET)

/person/{person_id}/info

GET /person/1/info returns:

```
{
  "name": "mike",
  "phone_number": "444-444-4444",
  "city": "Austin",
  "state": "TX",
  "street1": "213 South 1st St",
  "street2": "",
  "zip_code": "78704",
}
```

Task 2:

make a REST endpoint (POST) / Create

/person/create

POST /person/create accepts body:

```
{
  "name": "YOURNAME",
  "phone_number": "123-456-7890",
  "city": "Sacramento",
  "state": "CA",
  "street1": "112 Main St",
  "street2": "Apt 12",
  "zip_code": "12345",
}
```

returns 200