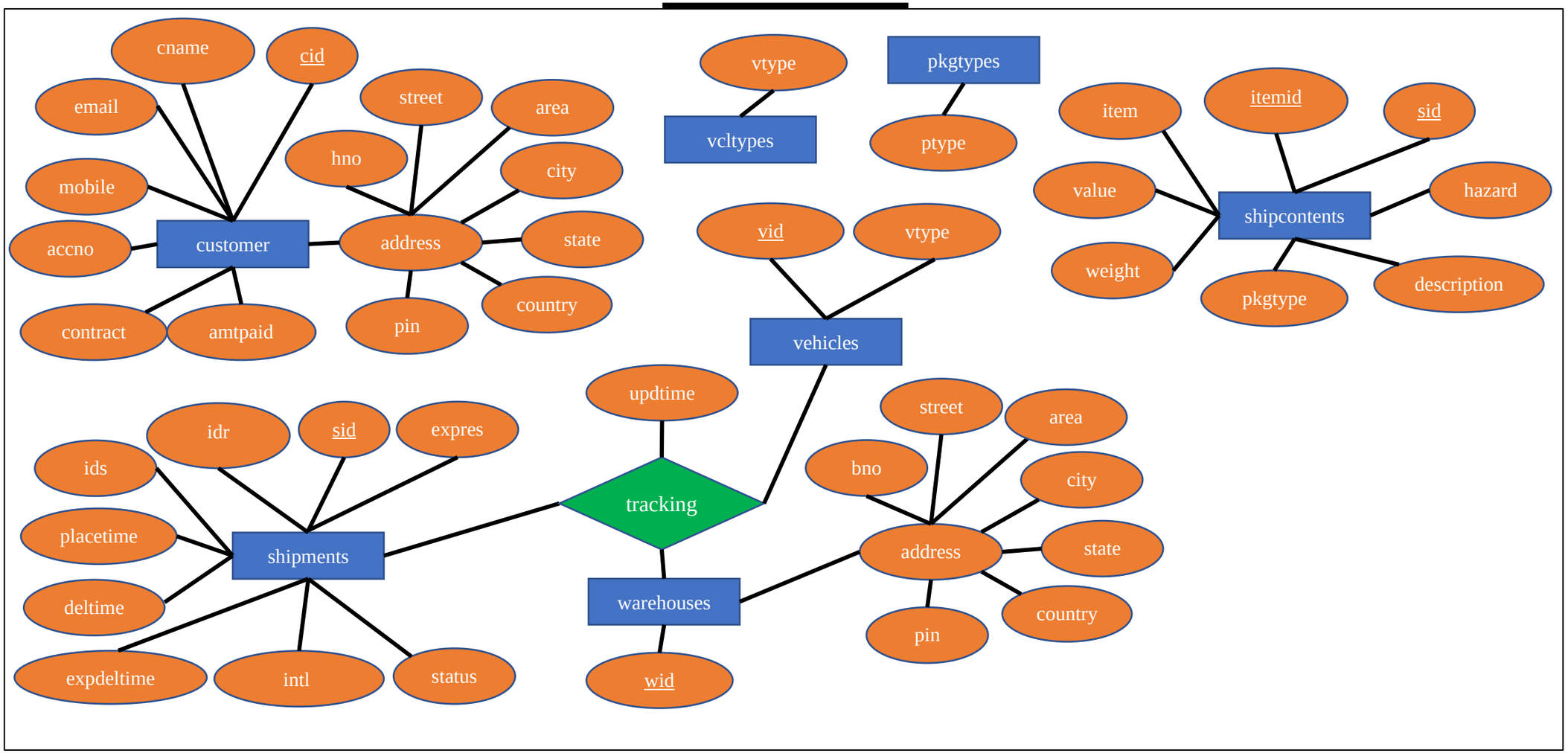
CS355 – Database Lab

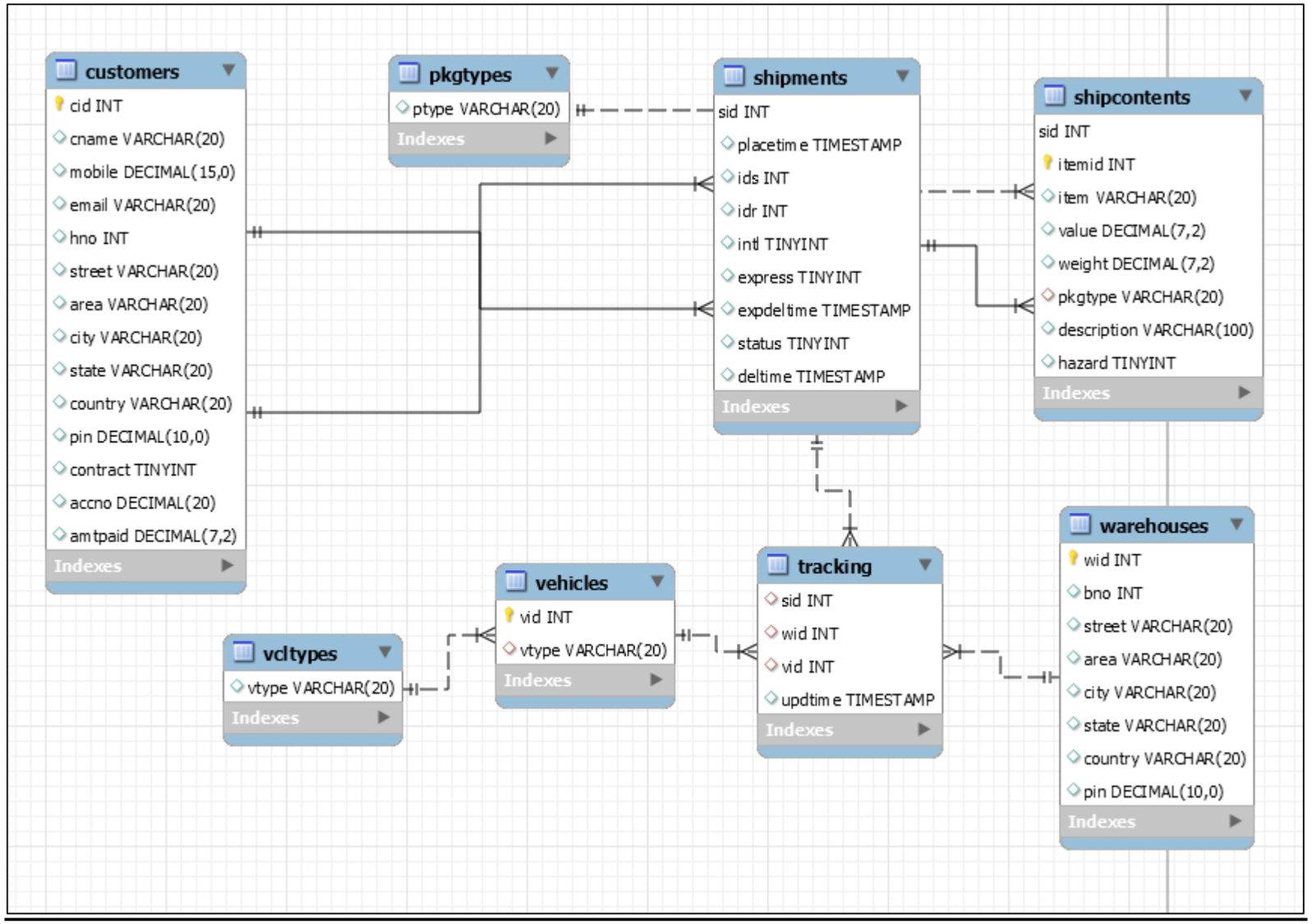
Name: M. Maheeth Reddy	**Roll No.**: 1801CS31	**Date**: 30 November 2020

Autumn 2020 – Mini Project Assignment

ER Model


cname
cid
email
street
area
hno
city
mobile
accno
customer
address
state
country
contract
amtpaid
pin
vtype
vcltypes
pkgtypes
ptype
item
itemid
sid
value
shipcontents
hazard
weight
pkgtype
description
vid
vtype
vehicles
updtime
idr
sid
expres
ids
placetime
shipments
deltime
expdeltime
intl
status
tracking
warehouses
wid
street
area
bno
city
address
state
pin
country

Relational Schema



Tables Present:

- customers
- pkgtypes
- shipments
- shipcontents
- vcltypes
- vehicles
- warehouses
- tracking

Description of each table:

customers

It stores the information about all customers such as name, contact details (mobile, email), address (house number, street, area, city, state, country, pincode), account number.
It also stores info whether a customer is having a contract with the company or not, and the amount they have paid so far.

Schema: customers (cid,cname,mobile,email,hno,street,area,city,state,country,pin,contract,accno,amtpaid)

MySQL Table:

```
-- create table for Customers
create table customers (
    cid int unsigned primary key,
    cname varchar(20),
    mobile decimal(15,0),
    email varchar(20),
    hno int unsigned,
    street varchar(20),
    area varchar(20),
    city varchar(20),
    state varchar(20),
    country varchar(20),
    pin decimal(10,0),
    contract tinyint default 0,
    accno decimal(20),
    amtpaid decimal(7,2) default 0,
    constraint validemail check (email regexp '^[A-Z0-9._%-]+@[A-Z0-9.-]+\.[A-Z]{2,4}$')
);
```

Here, cid is customer id,
cname is customer name,
mobile is contact number
email is email address of customer
hno is house number
and the rest of the attributes are street, area, city, state, country, pin of the address of customer
contract denotes whether customer is under contract with company or not. It takes values 1 or 0

pkgtypes

It stores all packaging types. The possible package types are flat envelope, small box, large box. Used for integrity purposes.

Schema: pkgtypes (ptype)

MySQL Code:

```
create table pkgtypes (
    ptype varchar(20) unique
);
```

Data:

```
insert into pkgtypes values ("flat_envelope"),("small_box"),("large_box");
```

shipments

It stores information about the shipments undertaken by company, like order place time, sender id, receiver id, international shipment or not, express delivery or not, expected delivery time, status of shipment, actual delivery time.

Schema: shipments (sid, placetime,ids,idr,intl,express,expdeltime,status,deltime)

MySQL Code:

```
create table shipments (
    sid int unsigned primary key,
    placetime timestamp,
    ids int unsigned,
    idr int unsigned,
    intl tinyint default 0,
    express tinyint default 0,
    expdeltime timestamp,
    status tinyint default 0,
    deltime timestamp default NULL,
    constraint valid_ids foreign key (ids) references customers(cid),
    constraint valid_idr foreign key (idr) references customers(cid)
);
```

Here, sid stands for shipment id,
place time represents order place time,
ids is sender id,
idr is receiver id,
intl shows whether shipment is international(if 1) or domestic(if 0),
express, if 0 represents faster delivery otherwise ordinary delivery,
expdeltime represents expected delivery date,
status shows whether shipping is finished if 1, or not if 0
deltime is actual delivery time, when shipment is delivered to receiver

shipcontents

It has details of items shipped like, item name, worth, weight, packaging type, description, if hazardous or not.

Schema: shipcontents (sid,itemid,item,value,weight,pkgtype,description,hazard)

MySQL Code:

```
create table shipcontents (
    sid int unsigned,
    itemid int unsigned,
    item varchar(20),
    value decimal(7,2),
    weight decimal(7,2),
    pkgtype varchar(20),
    description varchar(100),
    hazard tinyint,
    constraint valid_sid foreign key (sid) references shipments(sid),
    constraint valid_pkgtype foreign key (pkgtype) references pkgtypes(ptype),
    primary key (sid,itemid)
);
```

Here, sid is shipment id to which contents belong,
itemid is id of the item in shipment sid,
item is name of item being shipped,
value is worth of shipped item,
weight is weight of shipped item,
pkgtype represents type of packing of the item
description is an optional attribute, which is additional description
hazard represents whether the item is hazardous substance if 1, or not if 0

vcltypes

Contains all vehicle types present with company. For integrity purposes.

Schema: vcltypes (vtype)

MySQL Code:
```
create table vcltypes (
    vtype varchar(20) unique
);
```

Data: `insert into vcltypes values ("truck"),("minitruck"),("ship"),("plane");`

vehicles

contains info of vehicles like vehicle id and vehicle type.

Schema: vehicles (vid, vtype)

MySQL Code:
```
create table vehicles (
    vid int unsigned primary key,
    vtype varchar(20),
    constraint valid_vtype foreign key (vtype) references vcltypes(vtype)
);
```

warehouses

contains info of warehouses the company uses.

Schema: warehouses (wid,bno,street,area,city,state,country,pin)

MySQL Code:
```
create table warehouses (
    wid int unsigned primary key,
    bno int unsigned,
    street varchar(20),
    area varchar(20),
    city varchar(20),
    state varchar(20),
    country varchar(20),
    pin decimal(10,0)
);
```

Here, wid is warehouse id,
bno is building number,
and the rest of the attributes are street, area, city, state, country, pin of the address of the warehouse

tracking
contains info about shipment id, warehouse id, vehicle id and time of update of tracking information

Schema: tracking (sid,wid,vid,updtime)
```
create table tracking (
    sid int unsigned,
    wid int unsigned,
    vid int unsigned,
    updtime timestamp,
    constraint foreign key (sid) references shipments(sid),
    constraint foreign key (wid) references warehouses(wid),
    constraint foreign key (vid) references vehicles(vid)
);
```

Here, sid is shipment id,
wid is warehouse id to which the shipment is next taken to,
vid is vehicle id which takes shipment to warehouse wid,
updtime is time when the particular tracking update is done in the database

Queries Given

```
---- Assume a delivery truck (say truck no 1721) is destroyed in a crash.
-- Find all customers who had a package on that truck at the time of the crash.
select * from customers where cid in (select ids as cid from shipments where sid in (select sid from tracking
where vid=1721));

-- Find all recipients who had a package on that truck at the time of the crash.
select * from customers where cid in (select idr as cid from shipments where sid in (select sid from tracking
where vid=1721));

-- Find the last successful delivery by that truck prior to the crash.
select * from tracking natural join warehouses where vid=1721 order by updtime desc limit 1;


-- Find the customer who has shipped the most packages in the past year.
select * from customers where cid in (select ids as cid from (select ids,count(ids) as freq from shipments grou
p by ids order by freq desc limit 1) a);

-- Find the customer who has spent the most money on shipping in the past year through function and query
delimiter $$
create function amount_spent (tsid int unsigned)
returns decimal(7,2)
deterministic
begin
    declare tamtpaid decimal(7,2);
    declare sno,norows int;
    declare titemid int unsigned;
    declare cost decimal(7,2) default 0;
    declare tdeltime timestamp;

    select count(sid) into norows from shipcontents where sid=tsid;
    select amtpaid into tamtpaid from customers where cid=tsid;
    set sno=0;

    while(sno<norows) do
        select itemid from shipcontents where sid=tsid limit sno,1 into titemid;
        select deltime into tdeltime from shipments where sid=tsid;
        if(datediff(curdate(),tdeltime) < 365.25) then
            set cost = cost + bill_item(tsid,titemid);
            set sno=sno+1;
        end if;
    end while;

    return cost;
end$$
delimiter ;
select * from customers where amount_spent(cid) = (select max(amount_spent(sid)) as amt from shipments);


-- Find the street with the most customers.
select distinct street,area,city,state,country from customers where street in (select street from (select street,co
unt(street) as freq from customers group by street order by freq desc limit 1) a);

-- Find those packages that were not delivered within the promised time.
select * from shipments where deltime > expdeltime;

-- Function for bill listing charges by type of service.
delimiter $$
create function bill_item (tsid int unsigned,titemid int unsigned)
returns decimal(7,2)
deterministic
begin
    declare cost decimal(7,2) default 0;
    declare tintl tinyint;
```

```
    declare tpkgtype varchar(20);
    declare texpress tinyint;
    declare tweight decimal(7,2);

    select intl into tintl from shipments where sid=tsid;
    select pkgtype into tpkgtype from shipcontents where sid=tsid and itemid=titemid;
    select express into texpress from shipments where sid=tsid;
    select weight into tweight from shipcontents where sid=tsid and itemid=titemid;

    if(texpress) then
        set cost = cost + 50;
    end if;

    if(tintl) then
        if(tpkgtype="flat_envelope" or tpkgtype="small_box") then
            set cost = cost + 500;
        else
            set cost = cost + 5000;
            if(tweight >= 10) then
                set cost = cost + 500*(tweight-10)/10;
            end if;
        end if;
    else
        if(tpkgtype="flat_envelope" or tpkgtype="small_box") then
            set cost = cost + 200;
        else
            if(tweight < 10) then
                set cost = cost + 500;
            elseif(tweight < 20) then
                set cost = cost + 1000;
            else
                set cost = cost + 1500;
            end if;
        end if;
    end if;

    return (cost);
end$$
delimiter ;

-- Procedure for itemize billing listing each individual shipment and the charges for it.
delimiter $$
create procedure itemize_bill (in tsid int unsigned)
begin
    select *,bill_item(sid,itemid) as billing from shipcontents where sid=tsid;
end$$
delimiter ;

-
- Procedure which Takes Customer ID and provide the details such as customer name, address, and amount o
wed.
delimiter $$
create procedure amount_owed (in tsid int unsigned)
begin
    declare tamtpaid decimal(7,2);
    declare sno,norows int;
    declare titemid int unsigned;
    declare cost decimal(7,2) default 0;

    select count(sid) into norows from shipcontents where sid=tsid;
    select amtpaid into tamtpaid from customers where cid=tsid;
    set sno=0;

    while(sno<norows) do
        select itemid from shipcontents where sid=tsid limit sno,1 into titemid;
        set cost = cost + bill_item(tsid,titemid);
```

```
        set sno=sno+1;
    end while;

    select *,(cost-tamtpaid) as amount_owed from customers where cid=tsid;
end$$
delimiter ;
```

My Queries

```
-- Finding how customers are distributed across various countries
select country,count(country) as distribution from customers group by country;

-- Finding out if any hazardous materials have been shipped along with details
select * from shipcontents where hazard=1;
```