```
BEGIN;
CREATE TABLE IF NOT EXISTS public.bike
(
    id integer NOT NULL,
    typebike character varying(10)[] COLLATE pg_catalog."default" NOT NULL,
    numofpedal integer NOT NULL,
    numofrearseat integer NOT NULL,
    licenseplate character varying(20)[] COLLATE pg_catalog."default" NOT NULL,
    status character varying(10)[] COLLATE pg_catalog."default" NOT NULL,
    depositprice integer NOT NULL,
    dockid integer NOT NULL,
    CONSTRAINT bike_pkey PRIMARY KEY (id)
);

CREATE TABLE IF NOT EXISTS public.dock
(
    id integer NOT NULL,
    name character varying(45)[] COLLATE pg_catalog."default" NOT NULL,
    address character varying(200)[] COLLATE pg_catalog."default" NOT NULL,
    area integer NOT NULL,
    numofavailablespaces integer NOT NULL,
    numofavailablebikes integer NOT NULL,
    CONSTRAINT dock_pkey PRIMARY KEY (id)
);

CREATE TABLE IF NOT EXISTS public.ebike
(
    id integer NOT NULL,
    battery integer
);

CREATE TABLE IF NOT EXISTS public.invoice
(
    id integer NOT NULL,
    totalamount integer NOT NULL,
    rentid integer NOT NULL,
    CONSTRAINT invoice_pkey PRIMARY KEY (id)
);

CREATE TABLE IF NOT EXISTS public.paymenttransaction
(
    id integer NOT NULL,
    method character varying(10)[] COLLATE pg_catalog."default" NOT NULL,
    content character varying(200)[] COLLATE pg_catalog."default" NOT NULL,
    "createdAt" date NOT NULL,
    invoiceid integer NOT NULL,
    CONSTRAINT paymenttransaction_pkey PRIMARY KEY (id)
);
```

```
CREATE TABLE IF NOT EXISTS public.rent
(
    id integer NOT NULL,
    bikeid integer NOT NULL,
    rentalstarttime date NOT NULL,
    CONSTRAINT rent_pkey PRIMARY KEY (id)
);

ALTER TABLE IF EXISTS public.bike
    ADD CONSTRAINT fk_dockid_bike FOREIGN KEY (dockid)
    REFERENCES public.dock (id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;


ALTER TABLE IF EXISTS public.ebike
    ADD CONSTRAINT fk_bikeid_ebike FOREIGN KEY (id)
    REFERENCES public.bike (id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;


ALTER TABLE IF EXISTS public.invoice
    ADD CONSTRAINT fk_rentid_invoice FOREIGN KEY (rentid)
    REFERENCES public.rent (id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS unique_rent
    ON public.invoice(rentid);


ALTER TABLE IF EXISTS public.paymenttransaction
    ADD CONSTRAINT fk_invoiceid_payment FOREIGN KEY (invoiceid)
    REFERENCES public.invoice (id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS unique_invoice
    ON public.paymenttransaction(invoiceid);


ALTER TABLE IF EXISTS public.rent
    ADD CONSTRAINT fk_bikeid_rent FOREIGN KEY (bikeid)
    REFERENCES public.bike (id) MATCH SIMPLE
    ON UPDATE NO ACTION
    ON DELETE CASCADE;
CREATE INDEX IF NOT EXISTS unique_bikeid
    ON public.rent(bikeid);

END;
```