CREATE TABLE IF NOT EXISTS public."**User**"

(

user\_id integer NOT NULL DEFAULT nextval('"User\_user\_id\_seq"'::regclass),

name\_stud character varying(255) COLLATE pg\_catalog."default",

email character varying(255) COLLATE pg\_catalog."default",

university character varying(255) COLLATE pg\_catalog."default",

city character varying(255) COLLATE pg\_catalog."default",

age integer,

gender character varying(25) COLLATE pg\_catalog."default",

CONSTRAINT "User\_pkey" PRIMARY KEY (user\_id)

)

CREATE TABLE IF NOT EXISTS public."User **Preferences**"

(

preferences\_id integer NOT NULL DEFAULT nextval('"User Preferences\_preferences\_id\_seq"'::regclass),

user\_id integer,

notification\_preferences character varying(255) COLLATE pg\_catalog."default",

CONSTRAINT "User Preferences\_pkey" PRIMARY KEY (preferences\_id),

CONSTRAINT "User Preferences\_user\_id\_fkey" FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public.**expense**

(

expense\_id integer NOT NULL DEFAULT nextval('expense\_expense\_id\_seq'::regclass),

user\_id integer,

grocery numeric,

rent numeric,

personal numeric,

transportation numeric,

misc numeric,

total\_expense numeric,

CONSTRAINT expense\_pkey PRIMARY KEY (expense\_id),

CONSTRAINT expense\_user\_id\_fkey FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public.**budget**

(

budget\_id integer NOT NULL DEFAULT nextval('budget\_budget\_id\_seq'::regclass),

user\_id integer,

monthly\_budget numeric,

target\_saving numeric,

CONSTRAINT budget\_pkey PRIMARY KEY (budget\_id),

CONSTRAINT budget\_user\_id\_fkey FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public.**income**

(

income\_id integer NOT NULL DEFAULT nextval('income\_income\_id\_seq'::regclass),

user\_id integer,

income\_amount numeric,

CONSTRAINT income\_pkey PRIMARY KEY (income\_id),

CONSTRAINT income\_user\_id\_fkey FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public."Local Student Deals"

(

deal\_id integer NOT NULL DEFAULT nextval('"Local Student Deals\_deal\_id\_seq"'::regclass),

subway character varying(255) COLLATE pg\_catalog."default",

starbucks character varying(255) COLLATE pg\_catalog."default",

tacobell character varying(255) COLLATE pg\_catalog."default",

walmart character varying(255) COLLATE pg\_catalog."default",

user\_id integer,

CONSTRAINT "Local Student Deals\_pkey" PRIMARY KEY (deal\_id),

CONSTRAINT "Local Student Deals\_user\_id\_fkey" FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public."Emergency Fund"

(

    emergency\_fund\_id integer NOT NULL DEFAULT nextval('"Emergency Fund\_emergency\_fund\_id\_seq"'::regclass),

    user\_id integer,

    ef\_amount numeric,

    ef\_knowledge character varying(255) COLLATE pg\_catalog."default",

    CONSTRAINT "Emergency Fund\_pkey" PRIMARY KEY (emergency\_fund\_id),

    CONSTRAINT "Emergency Fund\_user\_id\_fkey" FOREIGN KEY (user\_id)

        REFERENCES public."User" (user\_id) MATCH SIMPLE

        ON UPDATE NO ACTION

        ON DELETE NO ACTION

)

 CREATE TABLE IF NOT EXISTS public."Academic Calendar Event"

(

    event\_id integer NOT NULL DEFAULT nextval('"Academic Calendar Event\_event\_id\_seq"'::regclass),

    user\_id integer,

    tuition\_fees\_due\_date date,

    insurance\_due\_date date,

    CONSTRAINT "Academic Calendar Event\_pkey" PRIMARY KEY (event\_id),

    CONSTRAINT "Academic Calendar Event\_user\_id\_fkey" FOREIGN KEY (user\_id)

        REFERENCES public."User" (user\_id) MATCH SIMPLE

        ON UPDATE NO ACTION

        ON DELETE NO ACTION

)

CREATE TABLE IF NOT EXISTS public."Part Time"

(

user\_id integer,

wage\_per\_hour numeric,

total numeric,

CONSTRAINT "Part Time\_user\_id\_fkey" FOREIGN KEY (user\_id)

REFERENCES public."User" (user\_id) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

)

-- Create a temporary table to store the counts for each priority value

CREATE TEMPORARY TABLE priority\_counts AS

SELECT

user\_id,

MAX(CASE WHEN subway = 'Priority 1' THEN 1 ELSE 0 END) AS subway\_p1,

MAX(CASE WHEN subway = 'Priority 2' THEN 1 ELSE 0 END) AS subway\_p2,

MAX(CASE WHEN subway = 'Priority 3' THEN 1 ELSE 0 END) AS subway\_p3,

MAX(CASE WHEN subway = 'Priority 4' THEN 1 ELSE 0 END) AS subway\_p4,

MAX(CASE WHEN starbucks = 'Priority 1' THEN 1 ELSE 0 END) AS starbucks\_p1,

MAX(CASE WHEN starbucks = 'Priority 2' THEN 1 ELSE 0 END) AS starbucks\_p2,

MAX(CASE WHEN starbucks = 'Priority 3' THEN 1 ELSE 0 END) AS starbucks\_p3,

MAX(CASE WHEN starbucks = 'Priority 4' THEN 1 ELSE 0 END) AS starbucks\_p4,

MAX(CASE WHEN tacobell = 'Priority 1' THEN 1 ELSE 0 END) AS tacobell\_p1,

MAX(CASE WHEN tacobell = 'Priority 2' THEN 1 ELSE 0 END) AS tacobell\_p2,

MAX(CASE WHEN tacobell = 'Priority 3' THEN 1 ELSE 0 END) AS tacobell\_p3,

MAX(CASE WHEN tacobell = 'Priority 4' THEN 1 ELSE 0 END) AS tacobell\_p4,

MAX(CASE WHEN walmart = 'Priority 1' THEN 1 ELSE 0 END) AS walmart\_p1,

MAX(CASE WHEN walmart = 'Priority 2' THEN 1 ELSE 0 END) AS walmart\_p2,

MAX(CASE WHEN walmart = 'Priority 3' THEN 1 ELSE 0 END) AS walmart\_p3,

MAX(CASE WHEN walmart = 'Priority 4' THEN 1 ELSE 0 END) AS walmart\_p4

FROM "Local Student Deals"

GROUP BY user\_id;

-- Create the discounts table

CREATE TABLE discounts AS

SELECT

CASE WHEN walmart\_p1 >= walmart\_p2 AND walmart\_p1 >= walmart\_p3 AND walmart\_p1 >= walmart\_p4 THEN 20

WHEN walmart\_p2 > walmart\_p1 AND walmart\_p2 >= walmart\_p3 AND walmart\_p2 >= walmart\_p4 THEN 15

WHEN walmart\_p3 > walmart\_p1 AND walmart\_p3 > walmart\_p2 AND walmart\_p3 >= walmart\_p4 THEN 10

ELSE 5 END AS walmart,

CASE WHEN subway\_p1 >= subway\_p2 AND subway\_p1 >= subway\_p3 AND subway\_p1 >= subway\_p4 THEN 20

WHEN subway\_p2 > subway\_p1 AND subway\_p2 >= subway\_p3 AND subway\_p2 >= subway\_p4 THEN 15

WHEN subway\_p3 > subway\_p1 AND subway\_p3 > subway\_p2 AND subway\_p3 >= subway\_p4 THEN 10

ELSE 5 END AS subway,

CASE WHEN starbucks\_p1 >= starbucks\_p2 AND starbucks\_p1 >= starbucks\_p3 AND starbucks\_p1 >= starbucks\_p4 THEN 20

WHEN starbucks\_p2 > starbucks\_p1 AND starbucks\_p2 >= starbucks\_p3 AND starbucks\_p2 >= starbucks\_p4 THEN 15

WHEN starbucks\_p3 > starbucks\_p1 AND starbucks\_p3 > starbucks\_p2 AND starbucks\_p3 >= starbucks\_p4 THEN 10

ELSE 5 END AS starbucks,

CASE WHEN tacobell\_p1 >= tacobell\_p2 AND tacobell\_p1 >= tacobell\_p3 AND tacobell\_p1 >= tacobell\_p4 THEN 20

WHEN tacobell\_p2 > tacobell\_p1 AND tacobell\_p2 >= tacobell\_p3 AND tacobell\_p2 >= tacobell\_p4 THEN 15

WHEN tacobell\_p3 > tacobell\_p1 AND tacobell\_p3 > tacobell\_p2 AND tacobell\_p3 >= tacobell\_p4 THEN 10

ELSE 5 END AS tacobell

FROM priority\_counts;

-- Create a temporary table to store the counts for each discount value and each store

CREATE TEMPORARY TABLE discount\_counts AS

SELECT

COUNT(CASE WHEN walmart = 20 THEN 1 END) AS walmart\_20\_count,

COUNT(CASE WHEN walmart = 15 THEN 1 END) AS walmart\_15\_count,

COUNT(CASE WHEN walmart = 5 THEN 1 END) AS walmart\_5\_count,

COUNT(CASE WHEN subway = 20 THEN 1 END) AS subway\_20\_count,

COUNT(CASE WHEN subway = 15 THEN 1 END) AS subway\_15\_count,

COUNT(CASE WHEN subway = 5 THEN 1 END) AS subway\_5\_count,

COUNT(CASE WHEN starbucks = 20 THEN 1 END) AS starbucks\_20\_count,

COUNT(CASE WHEN starbucks = 15 THEN 1 END) AS starbucks\_15\_count,

COUNT(CASE WHEN starbucks = 5 THEN 1 END) AS starbucks\_5\_count,

COUNT(CASE WHEN tacobell = 20 THEN 1 END) AS tacobell\_20\_count,

COUNT(CASE WHEN tacobell = 15 THEN 1 END) AS tacobell\_15\_count,

COUNT(CASE WHEN tacobell = 5 THEN 1 END) AS tacobell\_5\_count

FROM discounts;

-- Create the final discounts table

CREATE TABLE final\_discounts AS

SELECT

CASE WHEN walmart\_20\_count >= subway\_20\_count AND walmart\_20\_count >= starbucks\_20\_count AND walmart\_20\_count >= tacobell\_20\_count THEN 20

WHEN subway\_20\_count > walmart\_20\_count AND subway\_20\_count >= starbucks\_20\_count AND subway\_20\_count >= tacobell\_20\_count THEN 20

WHEN starbucks\_20\_count > walmart\_20\_count AND starbucks\_20\_count > subway\_20\_count AND starbucks\_20\_count >= tacobell\_20\_count THEN 20

ELSE 20 END AS walmart\_discount,

CASE WHEN walmart\_15\_count >= subway\_15\_count AND walmart\_15\_count >= starbucks\_15\_count AND walmart\_15\_count >= tacobell\_15\_count THEN 15

WHEN subway\_15\_count > walmart\_15\_count AND subway\_15\_count >= starbucks\_15\_count AND subway\_15\_count >= tacobell\_15\_count THEN 15

WHEN starbucks\_15\_count > walmart\_15\_count AND starbucks\_15\_count > subway\_15\_count AND starbucks\_15\_count >= tacobell\_15\_count THEN 15

ELSE 15 END AS subway\_discount,

CASE WHEN walmart\_5\_count >= subway\_5\_count AND walmart\_5\_count >= starbucks\_5\_count AND walmart\_5\_count >= tacobell\_5\_count THEN 5

WHEN subway\_5\_count > walmart\_5\_count AND subway\_5\_count >= starbucks\_5\_count AND subway\_5\_count >= tacobell\_5\_count THEN 5

WHEN starbucks\_5\_count > walmart\_5\_count AND starbucks\_5\_count > subway\_5\_count AND starbucks\_5\_count >= tacobell\_5\_count THEN 5

ELSE 5 END AS starbucks\_discount,

CASE WHEN walmart\_15\_count >= subway\_15\_count AND walmart\_15\_count >= starbucks\_15\_count AND walmart\_15\_count >= tacobell\_15\_count THEN 15

WHEN subway\_15\_count > walmart\_15\_count AND subway\_15\_count >= starbucks\_15\_count AND subway\_15\_count >= tacobell\_15\_count THEN 15

WHEN starbucks\_15\_count > walmart\_15\_count AND starbucks\_15\_count > subway\_15\_count AND starbucks\_15\_count >= tacobell\_15\_count THEN 15

ELSE 15 END AS tacobell\_discount

FROM discount\_counts;

query = """

SELECT g.lat, g.lng

FROM "User" u

JOIN geo g ON u.city = g.city;

"""