-- Table: public.User

-- DROP TABLE IF EXISTS public."User";

CREATE TABLE IF NOT EXISTS public."User"

(

username text COLLATE pg\_catalog."default" NOT NULL,

email text COLLATE pg\_catalog."default",

passwod text COLLATE pg\_catalog."default",

"firstName" text COLLATE pg\_catalog."default",

"lastName" text COLLATE pg\_catalog."default",

country text COLLATE pg\_catalog."default",

"registrationDat" date,

CONSTRAINT "User\_pkey" PRIMARY KEY (username)

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."User"

OWNER to postgres;

-- Table: public.Messages

-- DROP TABLE IF EXISTS public."Messages";

CREATE TABLE IF NOT EXISTS public."Messages"

(

"senderUsername" text COLLATE pg\_catalog."default" NOT NULL,

"receiverUsername" text COLLATE pg\_catalog."default" NOT NULL,

content text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone NOT NULL,

CONSTRAINT "Messages\_pkey" PRIMARY KEY ("senderUsername", "receiverUsername", "timestamp"),

CONSTRAINT user2\_fk FOREIGN KEY ("receiverUsername")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT user\_fk FOREIGN KEY ("senderUsername")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Messages"

OWNER to postgres;

-- Table: public.Befriends

-- DROP TABLE IF EXISTS public."Befriends";

CREATE TABLE IF NOT EXISTS public."Befriends"

(

"friend1Username" text COLLATE pg\_catalog."default" NOT NULL,

"friend2Username" text COLLATE pg\_catalog."default" NOT NULL,

"timestamp" timestamp without time zone,

CONSTRAINT "Befriends\_pkey" PRIMARY KEY ("friend1Username", "friend2Username"),

CONSTRAINT friend1\_fk FOREIGN KEY ("friend1Username")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT friend2\_fk FOREIGN KEY ("friend2Username")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Befriends"

OWNER to postgres;

-- Table: public.BlogEntry

-- DROP TABLE IF EXISTS public."BlogEntry";

CREATE TABLE IF NOT EXISTS public."BlogEntry"

(

"blogEntryID" integer NOT NULL,

username text COLLATE pg\_catalog."default",

content text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone,

votes integer,

CONSTRAINT "BlogEntry\_pkey" PRIMARY KEY ("blogEntryID"),

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."BlogEntry"

OWNER to postgres;

-- Table: public.Hashtag

-- DROP TABLE IF EXISTS public."Hashtag";

CREATE TABLE IF NOT EXISTS public."Hashtag"

(

"blogEntryID" integer NOT NULL,

tag text COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT "Hashtag\_pkey" PRIMARY KEY ("blogEntryID", tag),

CONSTRAINT "blogEntry\_fk" FOREIGN KEY ("blogEntryID")

REFERENCES public."BlogEntry" ("blogEntryID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Hashtag"

OWNER to postgres;

-- Table: public.Comments

-- DROP TABLE IF EXISTS public."Comments";

CREATE TABLE IF NOT EXISTS public."Comments"

(

"blogEntryID" integer NOT NULL,

username text COLLATE pg\_catalog."default" NOT NULL,

content text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone,

votes integer,

CONSTRAINT "Comments\_pkey" PRIMARY KEY ("blogEntryID", username),

CONSTRAINT "blogEntry\_fk" FOREIGN KEY ("blogEntryID")

REFERENCES public."BlogEntry" ("blogEntryID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Comments"

OWNER to postgres;

-- Table: public.Announcement

-- DROP TABLE IF EXISTS public."Announcement";

CREATE TABLE IF NOT EXISTS public."Announcement"

(

"announcementID" integer NOT NULL,

username text COLLATE pg\_catalog."default",

language text COLLATE pg\_catalog."default",

content text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone,

CONSTRAINT "Announcement\_pkey" PRIMARY KEY ("announcementID"),

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Announcement"

OWNER to postgres;

-- Table: public.Admin

-- DROP TABLE IF EXISTS public."Admin";

CREATE TABLE IF NOT EXISTS public."Admin"

(

username text COLLATE pg\_catalog."default" NOT NULL,

role text COLLATE pg\_catalog."default",

"contributionScore" integer,

CONSTRAINT "Admin\_pkey" PRIMARY KEY (username),

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Admin"

OWNER to postgres;

-- Table: public.Responsibilities

-- DROP TABLE IF EXISTS public."Responsibilities";

CREATE TABLE IF NOT EXISTS public."Responsibilities"

(

"adminUsername" text COLLATE pg\_catalog."default" NOT NULL,

responsibility text COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT "Responsibilities\_pkey" PRIMARY KEY ("adminUsername", responsibility),

CONSTRAINT admin\_fk FOREIGN KEY ("adminUsername")

REFERENCES public."Admin" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Responsibilities"

OWNER to postgres;

-- Table: public.ContestCreator

-- DROP TABLE IF EXISTS public."ContestCreator";

CREATE TABLE IF NOT EXISTS public."ContestCreator"

(

"creatorUsername" text COLLATE pg\_catalog."default" NOT NULL,

"assessmentScore" integer,

CONSTRAINT "ContestCreator\_pkey" PRIMARY KEY ("creatorUsername"),

CONSTRAINT user\_fk FOREIGN KEY ("creatorUsername")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."ContestCreator"

OWNER to postgres;

-- Table: public.Contestant

-- DROP TABLE IF EXISTS public."Contestant";

CREATE TABLE IF NOT EXISTS public."Contestant"

(

"contestantUsername" text COLLATE pg\_catalog."default" NOT NULL,

rating integer,

CONSTRAINT "Contestant\_pkey" PRIMARY KEY ("contestantUsername"),

CONSTRAINT user\_fk FOREIGN KEY ("contestantUsername")

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Contestant"

OWNER to postgres;

-- Table: public.Contest

-- DROP TABLE IF EXISTS public."Contest";

CREATE TABLE IF NOT EXISTS public."Contest"

(

"roundNumber" integer NOT NULL,

name text COLLATE pg\_catalog."default",

divsion integer,

"startTimestamp" timestamp without time zone,

length time without time zone,

description text COLLATE pg\_catalog."default",

CONSTRAINT "Contest\_pkey" PRIMARY KEY ("roundNumber")

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Contest"

OWNER to postgres;

-- Table: public.Organizes

-- DROP TABLE IF EXISTS public."Organizes";

CREATE TABLE IF NOT EXISTS public."Organizes"

(

"creatorUsername" text COLLATE pg\_catalog."default" NOT NULL,

"roundNumber" integer NOT NULL,

CONSTRAINT "Organizes\_pkey" PRIMARY KEY ("creatorUsername", "roundNumber"),

CONSTRAINT contest\_fk FOREIGN KEY ("roundNumber")

REFERENCES public."Contest" ("roundNumber") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT creator\_fk FOREIGN KEY ("creatorUsername")

REFERENCES public."ContestCreator" ("creatorUsername") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Organizes"

OWNER to postgres;

-- Table: public.CompetesIn

-- DROP TABLE IF EXISTS public."CompetesIn";

CREATE TABLE IF NOT EXISTS public."CompetesIn"

(

"contestantUsername" text COLLATE pg\_catalog."default" NOT NULL,

"roundNumber" integer NOT NULL,

score integer,

rank integer,

"timestampOfRegistration" timestamp without time zone,

CONSTRAINT "CompetesIn\_pkey" PRIMARY KEY ("contestantUsername", "roundNumber"),

CONSTRAINT contest\_fk FOREIGN KEY ("roundNumber")

REFERENCES public."Contest" ("roundNumber") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT contestant\_fk FOREIGN KEY ("contestantUsername")

REFERENCES public."Contestant" ("contestantUsername") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."CompetesIn"

OWNER to postgres;

-- Table: public.ProgrammingProblem

-- DROP TABLE IF EXISTS public."ProgrammingProblem";

CREATE TABLE IF NOT EXISTS public."ProgrammingProblem"

(

"problemID" integer NOT NULL,

"difficultyLevel" character(1) COLLATE pg\_catalog."default",

title text COLLATE pg\_catalog."default",

description text COLLATE pg\_catalog."default",

"timeLimit" integer,

"memoryLimit" integer,

"roundNumber" integer,

CONSTRAINT "ProgrammingProblem\_pkey" PRIMARY KEY ("problemID"),

CONSTRAINT contest\_fk FOREIGN KEY ("roundNumber")

REFERENCES public."Contest" ("roundNumber") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."ProgrammingProblem"

OWNER to postgres;

-- Table: public.Tag

-- DROP TABLE IF EXISTS public."Tag";

CREATE TABLE IF NOT EXISTS public."Tag"

(

"problemID" integer NOT NULL,

tag text COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT "Tag\_pkey" PRIMARY KEY ("problemID", tag),

CONSTRAINT problem\_fk FOREIGN KEY ("problemID")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Tag"

OWNER to postgres;

-- Table: public.TestCase

-- DROP TABLE IF EXISTS public."TestCase";

CREATE TABLE IF NOT EXISTS public."TestCase"

(

"problemID" integer NOT NULL,

"testCaseNumber" integer NOT NULL,

input text COLLATE pg\_catalog."default",

"expectedOutput" text COLLATE pg\_catalog."default",

CONSTRAINT "TestCase\_pkey" PRIMARY KEY ("problemID", "testCaseNumber"),

CONSTRAINT problem\_fk FOREIGN KEY ("problemID")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."TestCase"

OWNER to postgres;

-- Table: public.Solution

-- DROP TABLE IF EXISTS public."Solution";

CREATE TABLE IF NOT EXISTS public."Solution"

(

"problemID" integer NOT NULL,

"solutionID" integer NOT NULL,

"sourceCode" text COLLATE pg\_catalog."default",

CONSTRAINT "Solution\_pkey" PRIMARY KEY ("solutionID", "problemID"),

CONSTRAINT problem\_fk FOREIGN KEY ("problemID")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Solution"

OWNER to postgres;

-- Table: public.Notifies

-- DROP TABLE IF EXISTS public."Notifies";

CREATE TABLE IF NOT EXISTS public."Notifies"

(

"creatorUsername" text COLLATE pg\_catalog."default" NOT NULL,

"contestantUsername" text COLLATE pg\_catalog."default" NOT NULL,

"problemID" integer NOT NULL,

"notificationContent" text COLLATE pg\_catalog."default",

CONSTRAINT "Notifies\_pkey" PRIMARY KEY ("creatorUsername", "contestantUsername", "problemID"),

CONSTRAINT contestant\_fk FOREIGN KEY ("contestantUsername")

REFERENCES public."Contestant" ("contestantUsername") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT creator\_fk FOREIGN KEY ("creatorUsername")

REFERENCES public."ContestCreator" ("creatorUsername") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT "problemID" FOREIGN KEY ("problemID")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Notifies"

OWNER to postgres;

-- Table: public.Submission

-- DROP TABLE IF EXISTS public."Submission";

CREATE TABLE IF NOT EXISTS public."Submission"

(

"submissionID" integer NOT NULL,

"programmingLanguge" text COLLATE pg\_catalog."default",

"sourceCode" text COLLATE pg\_catalog."default",

verdict text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone,

CONSTRAINT "Submission\_pkey" PRIMARY KEY ("submissionID")

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Submission"

OWNER to postgres;

-- Table: public.Submits

-- DROP TABLE IF EXISTS public."Submits";

CREATE TABLE IF NOT EXISTS public."Submits"

(

username text COLLATE pg\_catalog."default" NOT NULL,

"submissionID" integer NOT NULL,

"problemNumber" integer NOT NULL,

CONSTRAINT "Submits\_pkey" PRIMARY KEY ("problemNumber", "submissionID", username),

CONSTRAINT problem\_fk FOREIGN KEY ("problemNumber")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT submission\_fk FOREIGN KEY ("submissionID")

REFERENCES public."Submission" ("submissionID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Submits"

OWNER to postgres;

-- Table: public.Hack

-- DROP TABLE IF EXISTS public."Hack";

CREATE TABLE IF NOT EXISTS public."Hack"

(

"hackID" integer NOT NULL,

test text COLLATE pg\_catalog."default",

verdict text COLLATE pg\_catalog."default",

"timestamp" timestamp without time zone,

CONSTRAINT "Hack\_pkey" PRIMARY KEY ("hackID")

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Hack"

OWNER to postgres;

-- Table: public.Hacks

-- DROP TABLE IF EXISTS public."Hacks";

CREATE TABLE IF NOT EXISTS public."Hacks"

(

"hackerUsername" text COLLATE pg\_catalog."default" NOT NULL,

"submissionID" integer NOT NULL,

"hackID" integer NOT NULL,

CONSTRAINT "Hacks\_pkey" PRIMARY KEY ("hackerUsername", "submissionID", "hackID"),

CONSTRAINT hack\_fk FOREIGN KEY ("hackID")

REFERENCES public."Hack" ("hackID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT hacker\_fk FOREIGN KEY ("hackerUsername")

REFERENCES public."Contestant" ("contestantUsername") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT submission\_fk FOREIGN KEY ("submissionID")

REFERENCES public."Submission" ("submissionID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Hacks"

OWNER to postgres;

-- Table: public.Battle

-- DROP TABLE IF EXISTS public."Battle";

CREATE TABLE IF NOT EXISTS public."Battle"

(

"battleID" integer NOT NULL,

"startTimestamp" timestamp without time zone,

"endTimestamp" timestamp without time zone,

CONSTRAINT "Battle\_pkey" PRIMARY KEY ("battleID")

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Battle"

OWNER to postgres;

-- Table: public.Joins

-- DROP TABLE IF EXISTS public."Joins";

CREATE TABLE IF NOT EXISTS public."Joins"

(

username text COLLATE pg\_catalog."default" NOT NULL,

"battleID" integer NOT NULL,

"isWinner" boolean,

CONSTRAINT "Joins\_pkey" PRIMARY KEY ("battleID", username),

CONSTRAINT battle\_fk FOREIGN KEY ("battleID")

REFERENCES public."Battle" ("battleID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID,

CONSTRAINT user\_fk FOREIGN KEY (username)

REFERENCES public."User" (username) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."Joins"

OWNER to postgres;

-- Table: public.ConsistsOf

-- DROP TABLE IF EXISTS public."ConsistsOf";

CREATE TABLE IF NOT EXISTS public."ConsistsOf"

(

"battleID" integer NOT NULL,

"problemID" integer NOT NULL,

CONSTRAINT "ConsistsOf\_pkey" PRIMARY KEY ("battleID", "problemID"),

CONSTRAINT battle\_fk FOREIGN KEY ("battleID")

REFERENCES public."Battle" ("battleID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT problem\_fk FOREIGN KEY ("problemID")

REFERENCES public."ProgrammingProblem" ("problemID") MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

NOT VALID

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public."ConsistsOf"

OWNER to postgres;