

University of Reading

Course: BSc. Computer Science

Module: Databases

Module Convenor: Dr. Frederic Stahl

Student: Vinícius de Oliveira Silva – **Student Number:** 24023627

Horse Racing Coursework Report

1. Introduction:

Having a solid and well planned database model is a vital feature for many computer based systems today. An efficient and effective database model can provide safety, stability and performance to the system as a whole, since it avoids unnecessary repetition of data and centralizes the data storage in a way that inconsistencies are much less likely to occur. Systems based on a centralized and efficient database model have a natural advantage when compared to others: all the information they need to access is stored on a single place in a structure that was specifically designed to handle the data in a way that will keep it consistent and reliable.

In this article, a database model will be proposed for keeping track of information about racehorses. The model will be discussed, analysed and after the final conclusions, it will be implemented using the PostgreSQL Database Management System.

2. Modelling:

2.1 Model Definition:

Analysing the assignment specification and considering the given scenario, it was possible to identify eight main entities that should exist in order to compose a satisfactory database model. Using the Codd-Date notation, it is possible to express the identified entities as following:

Relation: Horse Owner <u>ID</u> : Integer, Name: Varchar (30), *Phone: Character(11).	Relation: Stable <u>ID</u> : Integer, MaxCapacity: Integer, *Phone: Character(11).	Relation: Trainer <u>ID</u> : Integer, Name: Varchar (30), SuspectOfDoping: Boolean, *Phone: Character(11).
Relation: Jockey <u>ID</u> : Integer, Name: Varchar (30), *Phone: Character(11).	Relation: Racecourse *Phone: Character(11), Name: Varchar (30), TrackLength: Numeric (8,2), SeatingCapacity: Bigint.	Relation: Race <u>ID</u> : Integer, *RacecoursePhone: Character (11) Name: Varchar (30), Time: Timestamp, Public: Bigint, Prize: Money.
Relation: Horse <u>ID</u> : Integer, Name: Varchar (30), *StableID, *OwnerID *Sire: Integer, *Dam: Integer, Gender: Character, Height: Decimal, Weight: Decimal, Breed: Varchar	Relation: Contact Details <u>Phone</u> : Character Address: Text	

Table 1: Entities Definition

Obs.: Underlined attributes are Primary Keys and attributes marked with a “*” are Foreign Keys.

The relations between these entities are represented in the following diagram using the Barker-Ellis notation. It is important to notice that in order to represent many-to-many relations properly, it was necessary to add auxiliary intersection tables to the model.

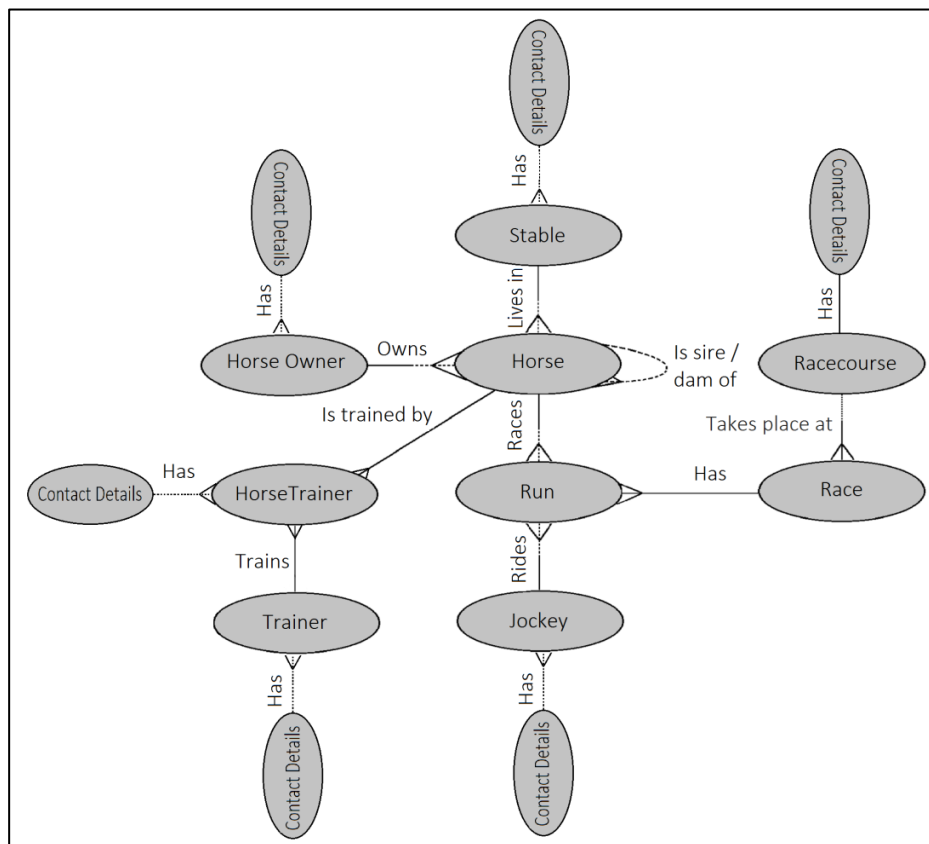


Image 1: E-R Diagram

Obs.: The entity “Contact Details” was represented several times in order to maintain the readability of the diagram.

The two extra intersection relations are defined as follows:

Relation: HorseTrainer <u>TrainerID</u> : Integer <u>HorseID</u> : Integer	Relation: Run <u>JockeyID</u> : Integer <u>HorseID</u> : Integer <u>RaceID</u> : Integer Placement: Integer
--	---

2.2 Constraints and Assumptions

Constraints and assumptions refer to the Business Rules of the system, which are the set of requirements that define the business policies and procedures. In order to bring the proposed model closer to real world systems, and at the same time, simplify the abstraction of the problem, constraints and assumptions were included in the project. The lists of constraints and assumptions are shown in the table below:

Constraints	Assumptions
1) Every horse must live in stable; 2) Every horse has an owner; 3) Every horse must have at least one trainer*; 4) Every trainer must train at least one horse*; 5) A race's public attendance cannot be greater than the respective racecourse seating capacity*; 6) Every run is associated with one race; 7) A male horse cannot be another horse's dam*; 8) A female horse cannot be another horse's sire*; 9) It is not possible to register a race that has not happened yet; 10) It is not possible to place a horse in a stable that is already full*; 11) A horse cannot be its own sire or dam; 12) A horse cannot be the sire or dam of any of its ancestors*; 13) A horse can only perform one run per race; 14) A jockey can only participate in one run per race; 15) A placement in a race must be unique. *Not implemented constraint, once it would require the use of triggers and/or stored procedures;	1) A horse can participate on a race regardless its gender; 2) A horse can participate on a race regardless its breed; 3) Every race has a winner. There are no draws; 4) Every Race Course is able to host any races; 5) A jockey is able to mount any horses. 6) A horse may have none or many children, but it can only have one sire and one dam.

Table 2: Constraints and Assumptions

2.3 Model Normalisation

In order to improve any given database model, it is very important to consider whether it is normalised or not. Normalisation is a method of stripping down each relation to its essential attributes, avoiding unnecessary storage repetition and loss of data.

Well-designed databases should be at least in third normal form and in order to show that the presented model is properly normalised, the normalisation rules will be stated and checked against the presented model.

2.3.1 First Normal Form

A relation is in First Normal Form if:

- **There are no repeating groups in the table (no multi-valued attributes):**

As it is possible to on Table 1, none of the used attributes is multi-valued;

- **All non-primary-key attributes are functionally dependent on the primary key (PK):**

In this model, except for the auxiliary intersection tables, all the Primary Keys Sets are composed by a single unique value. For this reason, every row on every table has a distinct primary key value, which means that all the non-primary key attributes are functionally dependent on the primary-key. In the intersection tables, although they have multiple values in their Primary-Key sets, care was taken to make sure that all the non-primary-key attributes are functionally dependent on the primary key set

2.3.2 Second Normal Form

A relation is in Second Normal Form if:

- **It is in 1NF:**

As it was shown on section 2.3.1, all the relations in the model are in First Normal Form;

- **Every non-primary-key attribute is fully functionally dependent on the primary key:**

Regarding all the tables in the model that have a single-attribute primary key, it is safe to state that they conform to this rule by definition. With respect to the intersection tables, care was taken to certify that only a combination of all the primary key attributes would functionally determine the other attributes in the relation.

2.3.3 Third Normal Form

A relation is in Third Normal Form if:

- **It is in 2NF:**

As it was shown on section 2.3.2, all the relations in the model are in Second Normal Form;

- **Every non-primary-key attribute depends only on the primary key:**

Analysing the attributes provided on Table 1, it is possible to notice that all the non-primary key attributes functionally depend only on the primary-key.

3. Implementation and Testing

3.1 Environment:

The environment used to implement the presented model consists in an Intel Core i7 based computer running PostgreSQL 9.4 under Windows 10 pro x64.

3.2 Implementation Details:

The designed model was physically implemented by using PostgreSQL Data Definition Language (DDL). Statements like "CREATE TABLE", "CREATE SEQUENCE", "ALTER TABLE" and "INSERT INTO" were extensively used while implementing the presented model. The full DDL script used to generate the tables and constraints as well as to insert data into the scheme is provided in the Appendix Section.

3.3 Testing

In order to test the model and show that it is able to meet the required features, some queries were executed on it and the obtained results are shown below:

- **Question 1:** Where has a given horse raced in the last 6 weeks?

Obs.: Assuming that referenced horse is the horse which code is 11 and that 6 weeks is equivalent to 42 days;

Query: `SELECT rcs.name FROM run rn INNER JOIN race rc ON rn.raceid = rc.id INNER JOIN racecourse rcs ON rc.racecoursephone = rcs.phone WHERE horseid=11 AND EXTRACT(DAY FROM (now() - rc.time)) < 42;`

Result:

Output pane	
Data Output	Explain Messages History
	name character varying(30)
1	Doncaster Racecourse

- **Question 2: Who has ridden the winners this year?**

Query: SELECT jk.name, rc.time, rn.placement FROM run rn INNER JOIN jockey jk ON rn.jockeyid = jk.id INNER JOIN race rc ON rn.raceid=rc.id WHERE rn.placement=1 AND EXTRACT (YEAR FROM rc.time) = EXTRACT (YEAR FROM now());

Result:

Output pane			
Data Output Explain Messages History			
	name character varying(30)	time timestamp without time zone	placement integer
1	Morgan Willians	2015-11-18 09:00:00	1
2	Albert Sterling	2015-08-15 09:00:00	1

- **Question 3: Produce a league table of winning trainers**

Query: SELECT tnr.name AS "Name", rc.name AS "Race", hrs.name AS "Winner Horse", to_char(date(rc.time), 'DD/MM/YYYY') AS "Date" FROM run rn INNER JOIN horsetrainer ht ON rn.horseid = ht.horseid INNER JOIN trainer tnr ON ht.trainerid = tnr.id INNER JOIN race rc ON rn.raceid = rc.id INNER JOIN horse hrs ON rn.horseid = hrs.id WHERE rn.placement =1 ORDER BY rc.time;

Result:

Output pane				
Data Output Explain Messages History				
	Trainer character varying(30)	Race character varying(30)	Winner Horse character varying(30)	Date text
1	William Smith	Ascott Horserace	Molly	21/05/2001
2	Johnathan Blewberg	Derby Horserace	Nikki	15/08/2014
3	Richard Fredman	Derby Horserace	Nikki	15/08/2014
4	Richard Fredman	St. Legers Horserace	Daisy	18/11/2014
5	Andy Dixon	Derby Horserace	Blaze	15/08/2015
6	Ronald MdCrank	St. Legers Horserace	Champ	18/11/2015
7	Andy Dixon	St. Legers Horserace	Champ	18/11/2015

- **Question 4: Which trainers are suspected of doping?**

Query: SELECT tnr.name AS "Trainer Name", text(suspectofdoping) AS "Suspect of Doping" FROM trainer tnr WHERE suspectofdoping = true;

Result:

Output pane		
Data Output Explain Messages History		
	Trainer Name character varying(30)	Suspect of Doping text
1	Andy Dixon	true
2	William Smith	true

- **Question 5: List all winners of the Derby and the St. Leger in the same year**

Query: SELECT jky.name AS "Jockey Name", rc.name AS "Race 1 name", rc.time AS "Race 1 time", rc.prize AS "Race 1 prize", rc2.name AS "Race 2 Name", rc2.time AS "Race 2 time", rc2.prize AS "Race 2 prize" FROM run rn INNER JOIN race rc ON rn.raceid = rc.id INNER JOIN run rn2 ON rn.jockeyid = rn2.jockeyid INNER JOIN race rc2 ON rn2.raceid = rc2.id INNER JOIN jockey jky ON rn.jockeyid = jky.id WHERE rn.placement=1 AND EXTRACT(YEAR FROM rc.time) = EXTRACT(YEAR FROM rc2.time) AND (((rc.name ='St. Legers Horserace' AND rc2.name = 'Derby Horserace')) OR ((rc.name ='St. Legers Horserace' AND rc2.name = 'Derby Horserace'))) ;

Result:

Output pane							
Data Output Explain Messages History							
	Jockey Name character varying(30)	Race 1 name character varying(30)	Race 1 time timestamp without time zone	Race 1 prize money	Race 2 Name character varying(30)	Race 2 time timestamp without time zone	Race 2 prize money
1	Morgan Willians	St. Legers Horserace	2014-11-18 09:00:00	£4,500.00	Derby Horserace	2014-08-15 09:00:00	£5,000.00

- **Question 6:** Find the pedigree (to 4 generations) of a given horse.

Obs.: The presented result shows the pedigree to 4 generations of an arbitrary horse, in this case, “Poncho”, which ID is 19.

Query: SELECT horsename AS "Horse Name", horsebreed AS "Horse Breed", COALESCE(sirebreed,'Unknown') AS "Sire Breed", COALESCE(dambreed,'Unknown') AS "Dam Breed", COALESCE(grandsirebreed, 'Unknown') AS "Grandsire Breed", COALESCE(granddambreed, 'Unknown') AS "Granddam Breed", COALESCE(greatgrandsirebreed, 'Unknown') AS "Great-grandsire Breed", COALESCE(greatgranddambreed, 'Unknown') AS "Great-granddam Breed" FROM ((SELECT gen04.id AS ID1, gen04.name AS horsename, gen04.breed AS horsebreed, gen03.breed AS dambreed, gen02.breed AS granddambreed, gen01.breed AS greatgranddambreed FROM horse gen04 LEFT JOIN horse gen03 ON gen04.dam = gen03.id LEFT JOIN horse gen02 ON gen03.dam = gen02.id LEFT JOIN horse gen01 ON gen02.dam = gen01.id) t1 INNER JOIN (SELECT gen14.id AS ID2, gen13.breed AS sirebreed, gen12.breed AS grandsirebreed, gen11.breed AS greatgrandsirebreed FROM horse gen14 LEFT JOIN horse gen13 ON gen14.sire = gen13.id LEFT JOIN horse gen12 ON gen13.sire = gen12.id LEFT JOIN horse gen11 ON gen12.sire = gen11.id) t2 ON ID1 = ID2) WHERE ID1 = 19;

Result:

Output pane								
Data Output Explain Messages History								
	Horse Name character varying(30)	Horse Breed character varying(30)	Sire Breed character varying	Dam Breed character varying	Grandsire Breed character varying	Granddam Breed character varying	Great-grandsire Breed character varying	Great-granddam Breed character varying
1	Poncho	Shetland Oldenburg	Shetland Oldenburg	Oldenburg	Shetland Pony	American Oldenburg	Mustang	American Saddlebred

- **Question 7:** Who has ridden both a horse and its child to a top-three place in any race?

Query: SELECT j.id, j.name FROM run r INNER JOIN horse h ON r.horseid=h.id INNER JOIN jockey j ON r.jockeyid = j.id WHERE placement <= 3 AND ((h.sire IN (SELECT horseid FROM run r2 WHERE r2.placement <= 3 AND r2.jockeyid = r.jockeyid)) OR (h.dam IN (SELECT horseid FROM run r2 WHERE r2.placement <= 3 AND r2.jockeyid = r.jockeyid)));

Result:

	id integer	name character varying(30)
1	1	Richard Friederich
2	4	Harold Mertens

4. Conclusions

The development of the presented coursework has occurred as expected, with no major issues. However, it is important to state that some changes had to be made in the database model during the normalisation process, which shows that even if designed considering the normalisation rules, the model should be checked for eventual features that not conform to these rules.

In general, it can be said that the coursework development was satisfactory even though the coursework specification was not very precise. Many assumptions were needed and intense use of creativity was necessary in order to design the presented model.

In conclusion, the process of development of the model was a very good opportunity for us to have a first contact with database design and to practice the skills that were taught during the term.

5. References:

<http://whatis.techtarget.com/definition/business-rule> - Accessed on 15/11/2015

6. Appendix

Database Generation Script:

```
CREATE TABLE contactdetails (  
    phone character(11) NOT NULL,  
    address text  
);  
  
CREATE TABLE horse (  
    id integer NOT NULL,  
    name character varying(30),  
    sire integer,  
    dam integer,  
    gender character(1),  
    height numeric(3,2),  
    weight numeric(6,3),  
    breed character varying(30),  
    stableid integer NOT NULL,  
    ownerid integer NOT NULL,  
    CONSTRAINT own_dam CHECK ((dam <> id)),  
    CONSTRAINT own_sire CHECK ((sire <> id))  
);  
  
CREATE SEQUENCE horse_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;  
  
CREATE TABLE horseowner (  
    id integer NOT NULL,  
    name character varying(30),  
    phone character(11)  
);  
  
CREATE SEQUENCE horseowner_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;  
  
CREATE TABLE horsetrainer (  
    trainerid integer NOT NULL,  
    horseid integer NOT NULL  
);  
  
CREATE TABLE jockey (  
    id integer NOT NULL,  
    name character varying(30),  
    phone character(11)  
);  
  
CREATE SEQUENCE jockey_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```

CREATE TABLE race (
    id integer NOT NULL,
    racecoursephone character(11),
    name character varying(30),
    "time" timestamp without time zone NOT NULL,
    public bigint,
    prize money,
    CONSTRAINT no_future_races CHECK ((now() > "time"))
);

CREATE SEQUENCE race_id_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

CREATE TABLE racecourse (
    phone character(11) NOT NULL,
    name character varying(30),
    tracklength numeric(8,2),
    seatingcapacity bigint
);

CREATE TABLE run (
    jockeyid integer NOT NULL,
    horseid integer NOT NULL,
    raceid integer NOT NULL,
    placement integer
);

CREATE TABLE stable (
    id integer NOT NULL,
    maxcapacity integer,
    phone character(11)
);

CREATE SEQUENCE stable_id_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

CREATE TABLE trainer (
    id integer NOT NULL,
    name character varying(30),
    suspectofdoping boolean DEFAULT false NOT NULL,
    phone character(11)
);

CREATE SEQUENCE trainer_id_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE horse ALTER COLUMN id SET DEFAULT nextval('horse_id_seq'::regclass);

ALTER TABLE horseowner ALTER COLUMN id SET DEFAULT
nextval('horseowner_id_seq'::regclass);

```



```
ALTER      TABLE      jockey      ALTER      COLUMN      id      SET      DEFAULT
nextval('jockey_id_seq'::regclass);
```

```
ALTER TABLE race ALTER COLUMN id SET DEFAULT nextval('race_id_seq'::regclass);
```

```
ALTER      TABLE      stable      ALTER      COLUMN      id      SET      DEFAULT
nextval('stable_id_seq'::regclass);
```

```
ALTER      TABLE      trainer      ALTER      COLUMN      id      SET      DEFAULT
nextval('trainer_id_seq'::regclass);
```

```
INSERT INTO contactdetails (phone, address) VALUES ('07464240490', 'RG1 1TG
Queen Victoria Street, Reading, Berkshire, UK');
INSERT INTO contactdetails (phone, address) VALUES ('02089468579', ' 24 a/b High
Street, Wimbledon, London SW19 5DX');
INSERT INTO contactdetails (phone, address) VALUES ('01792390554', ' The Poplars
Pitton Cross, Rhossili, Swansea SA3 1PH');
INSERT INTO contactdetails (phone, address) VALUES ('01691271840', 'Springhill
farm, Selattyn, Oswestry, Shropshire SY10 7NZ');
INSERT INTO contactdetails (phone, address) VALUES ('01372464348', 'Portsmouth
Rd,Esher, Surrey KT10 9AJ');
INSERT INTO contactdetails (phone, address) VALUES ('01243755022',
'Goodwood,Chichester, West Sussex PO18 0PS');
INSERT INTO contactdetails (phone, address) VALUES ('07490264090', 'St.
George''s Hall, Oakhurst, Room 59 - RG1 5HZ, Reading, UK');
INSERT INTO contactdetails (phone, address) VALUES ('02087610770', 'SE21 8LG,
Rosendale Rd., London, UK');
INSERT INTO contactdetails (phone, address) VALUES ('02057890550', 'SW9 7NH
Wiltshire Road, London, UK');
INSERT INTO contactdetails (phone, address) VALUES ('07932543900', 'L7 7EU,
Little Canning St., Liverpool, UK');
INSERT INTO contactdetails (phone, address) VALUES ('01344878501', 'Ascott,
Berkshire, SL5 7JX');
INSERT INTO contactdetails (phone, address) VALUES ('61395181362', '591-659
Princes Highway,Springvale VIC 3171, Australia');
INSERT INTO contactdetails (phone, address) VALUES ('01302304200', 'The
Grandstand,Leger Way,Doncaster DN2 6BB');
INSERT INTO contactdetails (phone, address) VALUES ('01372726311', 'Epsom Downs,
Surrey KT18 5LQ, UK');
```

```
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (1, 'Bolt', NULL, NULL, 'M', 2.30, 302.614, 'Mustang', 2, 4);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (2, 'Daisy', NULL, NULL, 'F', 1.98, 280.410, 'American
Saddlebred', 2, 3);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (4, 'Milo', 1, 2, 'M', 2.05, 250.000, 'Shetland Pony', 3, 6);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (9, 'Coco', 6, 2, 'F', 1.98, 250.218, 'Belgian SaddleBred', 3,
5);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (6, 'Blaze', 1, 2, 'M', 1.94, 274.358, 'Belgian Horse', 4, 2);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (11, 'Champ', 6, 9, 'M', 1.95, 275.900, 'Belgian Oldenburg', 3,
5);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (12, 'Molly', 4, 9, 'F', 1.88, 240.650, 'SaddleBred Pony', 4,
2);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (13, 'Luna', 11, 2, 'F', 1.97, 262.786, 'American Oldenburg', 2,
3);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (14, 'Buck', 11, 12, 'M', 1.98, 282.573, 'Saddlebred Oldenburg',
4, 5);
```

```

INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (15, 'Nikki', 14, 13, 'F', 1.90, 270.954, 'Oldenburg', 2, 3);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (19, 'Poncho', 17, 15, 'M', 1.92, 274.861, 'Shetland Oldenburg',
1, 2);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (17, 'Scout', 4, 13, 'M', 1.97, 280.970, 'Shetland Oldenburg',
1, 4);
INSERT INTO horse (id, name, sire, dam, gender, height, weight, breed, stableid,
ownerid) VALUES (20, 'Dixie', 19, 12, 'F', 1.90, 269.526, 'Shetland Saddlebred',
3, 6);

INSERT INTO horseowner (id, name, phone) VALUES (2, 'Michael K. Cornwall',
'01243755022');
INSERT INTO horseowner (id, name, phone) VALUES (3, 'Sarah K. Cornwall',
'01243755022');
INSERT INTO horseowner (id, name, phone) VALUES (4, 'Marco P. McCriday',
'07932543900');
INSERT INTO horseowner (id, name, phone) VALUES (5, 'Savannah H. Collins',
'02087610770');
INSERT INTO horseowner (id, name, phone) VALUES (6, 'Richard K. Ruffle',
'07464240490');

INSERT INTO horsetrainer (trainerid, horseid) VALUES (1, 1);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (1, 4);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (2, 6);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (2, 13);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (3, 2);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (3, 9);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (4, 11);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (5, 12);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (4, 20);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (5, 13);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (1, 14);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (6, 15);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (6, 19);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (2, 11);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (4, 17);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (4, 1);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (3, 15);
INSERT INTO horsetrainer (trainerid, horseid) VALUES (5, 19);

INSERT INTO jockey (id, name, phone) VALUES (1, 'Richard Friederich',
'01691271840');
INSERT INTO jockey (id, name, phone) VALUES (3, 'Morgan Willians',
'02057890550');
INSERT INTO jockey (id, name, phone) VALUES (4, 'Harold Mertens',
'02087610770');
INSERT INTO jockey (id, name, phone) VALUES (5, 'Vinicius Silva',
'07490264090');
INSERT INTO jockey (id, name, phone) VALUES (6, 'Jason Collins', '07932543900');
INSERT INTO jockey (id, name, phone) VALUES (7, 'Albert Sterling',
'01792390554');
INSERT INTO jockey (id, name, phone) VALUES (8, 'Heather Friars',
'07464240490');
INSERT INTO jockey (id, name, phone) VALUES (9, 'Felicia Morgans',
'01691271840');

INSERT INTO race (id, racecoursephone, name, "time", public, prize) VALUES (3,
'01344878501', 'Ascott Horserace', '2001-05-21 00:00:00', 8000, '£4,000.00');
INSERT INTO race (id, racecoursephone, name, "time", public, prize) VALUES (5,
'01302304200', 'St. Legers Horserace', '2015-11-18 09:00:00', 17486,
'£5,000.00');

```

```

INSERT INTO race (id, racecoursephone, name, "time", public, prize) VALUES (7,
'01302304200', 'St. Legers Horserace', '2014-11-18 09:00:00', 18000,
'£4,500.00');
INSERT INTO race (id, racecoursephone, name, "time", public, prize) VALUES (9,
'01372726311', 'Derby Horserace', '2015-08-15 09:00:00', 10000, '£6,000.00');
INSERT INTO race (id, racecoursephone, name, "time", public, prize) VALUES (10,
'01372726311', 'Derby Horserace', '2014-08-15 09:00:00', 9000, '£5,000.00');

INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('01344878501', 'Ascott Racecourse', 500.00, 10000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('02089468579', 'Wimbledon Racecourse', 600.00, 15000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('61395181362', 'Royal National Racecourse', 550.00, 8000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('02057890550', 'Royal National Racecourse', 700.00, 20000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('01691271840', 'Springhill Racecourse', 300.00, 1000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('01302304200', 'Doncaster Racecourse', 500.00, 18000);
INSERT INTO racecourse (phone, name, tracklength, seatingcapacity) VALUES
('01372726311', 'Epsom Downs Racecourse', 400.00, 15000);

INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (5, 1, 3, 2);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (7, 12, 3, 1);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (4, 19, 5, 3);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (8, 13, 5, 2);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (3, 11, 5, 1);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (9, 4, 7, 3);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (6, 9, 7, 2);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (3, 2, 7, 1);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (4, 17, 9, 3);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (1, 11, 9, 2);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (7, 6, 9, 1);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (5, 12, 10, 3);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (3, 15, 10, 1);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (8, 19, 10, 2);
INSERT INTO run (jockeyid, horseid, raceid, placement) VALUES (1, 14, 3, 3);

INSERT INTO stable (id, maxcapacity, phone) VALUES (1, 25, '02089468579');
INSERT INTO stable (id, maxcapacity, phone) VALUES (2, 50, '01372464348');
INSERT INTO stable (id, maxcapacity, phone) VALUES (3, 10, '01243755022');
INSERT INTO stable (id, maxcapacity, phone) VALUES (4, 30, '01792390554');

INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (1, 'Julie Dixon',
false, '02057890550');
INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (2, 'Andy Dixon',
true, '02057890550');
INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (3, 'Richard
Fredman', false, '07464240490');
INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (4, 'Ronald
MdCrank', false, '01792390554');
INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (5, 'William
Smith', true, '01691271840');
INSERT INTO trainer (id, name, suspectofdoping, phone) VALUES (6, 'Johnathan
Blewberg', false, '07932543900');

ALTER TABLE contactdetails
    ADD CONSTRAINT contact_pk PRIMARY KEY (phone);

ALTER TABLE horse
    ADD CONSTRAINT horse_id PRIMARY KEY (id);

ALTER TABLE horsetrainer
    ADD CONSTRAINT horse_trainer_pk PRIMARY KEY (trainerid, horseid);

```

```

ALTER TABLE jockey
    ADD CONSTRAINT jockey_id PRIMARY KEY (id);

ALTER TABLE run
    ADD CONSTRAINT one_run_per_horse_per_race UNIQUE (horseid, raceid);

ALTER TABLE run
    ADD CONSTRAINT one_run_per_jockey_per_race UNIQUE (jockeyid, raceid);

ALTER TABLE horseowner
    ADD CONSTRAINT owner_pk PRIMARY KEY (id);

ALTER TABLE race
    ADD CONSTRAINT race_id PRIMARY KEY (id);
ALTER TABLE racecourse
    ADD CONSTRAINT racecourse_pk PRIMARY KEY (phone);

ALTER TABLE run
    ADD CONSTRAINT run_pk PRIMARY KEY (jockeyid, horseid, raceid);

ALTER TABLE stable
    ADD CONSTRAINT stable_id PRIMARY KEY (id);

ALTER TABLE trainer
    ADD CONSTRAINT trainer_id PRIMARY KEY (id);

ALTER TABLE run
    ADD CONSTRAINT unique_placement_per_race UNIQUE (raceid, placement);

ALTER TABLE horse
    ADD CONSTRAINT horse_dam FOREIGN KEY (dam) REFERENCES horse(id) ON UPDATE
    CASCADE ON DELETE SET NULL;

ALTER TABLE horse
    ADD CONSTRAINT horse_owner FOREIGN KEY (ownerid) REFERENCES horseowner(id)
    ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE horse
    ADD CONSTRAINT horse_sire FOREIGN KEY (sire) REFERENCES horse(id) ON UPDATE
    CASCADE ON DELETE SET NULL;

ALTER TABLE horse
    ADD CONSTRAINT horse_stable FOREIGN KEY (stableid) REFERENCES stable(id) ON
    UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE horsetrainer
    ADD CONSTRAINT horsetrainer_horseid FOREIGN KEY (horseid) REFERENCES
    horse(id) ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE horsetrainer
    ADD CONSTRAINT horsetrainer_trainerid FOREIGN KEY (trainerid) REFERENCES
    trainer(id) ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE jockey
    ADD CONSTRAINT jockey_phone FOREIGN KEY (phone) REFERENCES
    contactdetails(phone) ON UPDATE CASCADE ON DELETE SET NULL;

ALTER TABLE horseowner
    ADD CONSTRAINT owner_phone FOREIGN KEY (phone) REFERENCES
    contactdetails(phone) ON UPDATE CASCADE ON DELETE SET NULL;

ALTER TABLE race
    ADD CONSTRAINT race_racecourse FOREIGN KEY (racecoursephone) REFERENCES
    racecourse(phone) ON UPDATE CASCADE ON DELETE SET NULL;

```

```
ALTER TABLE racecourse
    ADD CONSTRAINT racecourse_phone FOREIGN KEY (phone) REFERENCES
contactdetails(phone) ON UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE run
    ADD CONSTRAINT run_horse FOREIGN KEY (horseid) REFERENCES horse(id) ON
UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE run
    ADD CONSTRAINT run_jockey FOREIGN KEY (jockeyid) REFERENCES jockey(id) ON
UPDATE CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE run
    ADD CONSTRAINT run_race FOREIGN KEY (raceid) REFERENCES race(id) ON UPDATE
CASCADE ON DELETE CASCADE;
```

```
ALTER TABLE stable
    ADD CONSTRAINT stable_phone FOREIGN KEY (phone) REFERENCES
contactdetails(phone) ON UPDATE CASCADE ON DELETE SET NULL;
```

```
ALTER TABLE trainer
    ADD CONSTRAINT trainer_phone FOREIGN KEY (phone) REFERENCES
contactdetails(phone) ON UPDATE CASCADE ON DELETE SET NULL;
```