```
 1
 2  /*
 3  ** JOEBASE - The Action Figure Database
 4  ** joebase-create-db: This file created the primary database table structures for Joebase.
 5  **
 6  ** Copyright Leonard Armstrong, 2018
 7  ** IST659
 8  */
 9
10  USE joebase;
11  GO
12
13  ----------------------------------------------------------------------------------------------
14  -- DROP Section
15  -- Ensure the database is cleared of any residual data, tables, etc. before recreating.
16  ----------------------------------------------------------------------------------------------
17
18  DROP TABLE IF EXISTS product_product_image;
19  GO
20
21  DROP TABLE IF EXISTS action_figure_genre;
22  GO
23
24  DROP TABLE IF EXISTS action_figure;
25  GO
26
27  DROP TABLE IF EXISTS product_series;
28  GO
29
30  DROP TABLE IF EXISTS product_genre
31  GO
32
33  DROP TABLE IF EXISTS product;
34  GO
35
36  DROP TABLE IF EXISTS product_type;
37  GO
```

```sql
38
39  DROP TABLE IF EXISTS product_image;
40  GO
41
42  DROP TABLE IF EXISTS genre;
43  GO
44
45  DROP TABLE IF EXISTS series;
46  GO
47
48  DROP TABLE IF EXISTS retailer;
49  GO
50
51  DROP TABLE IF EXISTS manufacturer;
52  GO
53
54  --------------------------------------------------------------------------------------
55  -- CREATE Section
56  -- Create all tables and table constraints requrired for Joebase
57  --------------------------------------------------------------------------------------
58
59  -- manufacturer: Lookup table for all producers of action figure products.
60  CREATE TABLE manufacturer
61  (
62      manufacturer_id     INT IDENTITY PRIMARY KEY,        -- DB identifier of record
63      manufacturer_name   NVARCHAR(128) UNIQUE NOT NULL    -- Human identifier of record
64  );
65  GO
66
67  -- retailer: Lookup table for all retailers of action figure products.
68  CREATE TABLE retailer
69  (
70      retailer_id     INT IDENTITY PRIMARY KEY,        -- DB identifier of record
71      retailer_name   NVARCHAR(128) UNIQUE NOT NULL,   -- Human identifier of record
72      retailer_url    VARCHAR(2048)                    -- Online presence of retailer
73  );
74  GO
```

```
 75
 76   -- Primary classification system of action figure products according to the theme(s) they portray
 77   CREATE TABLE genre
 78   (
 79       genre_id    INT IDENTITY PRIMARY KEY,                -- DB identifier of record
 80       genre_name  NVARCHAR(128) UNIQUE NOT NULL            -- Human identifier of record
 81   );
 82   GO
 83
 84   -- Elexctronic images of action figure products
 85   CREATE TABLE product_image
 86   (
 87       product_image_id            INT IDENTITY PRIMARY KEY,
 88                                                    -- DB identifier of record
 89       url                         VARCHAR(2048) NOT NULL, -- Electronic "home" of image
 90       product_image_description   NVARCHAR(1024),        -- Human-readable description of image content
 91       product_image_credit        NVARCHAR(1024)         -- Acknowledgement of image producer/owner/copyright holder.
 92   );
 93   GO
 94
 95   -- Type classification of products (scale, figure vs. accessory set, vehicle, etc.)
 96   CREATE TABLE product_type
 97   (
 98       product_type_id         INT IDENTITY PRIMARY KEY,   -- DB identifier of record
 99       product_type_name       VARCHAR(128) UNIQUE NOT NULL,
100                                                    -- Human identifier of record
101       product_type_description NVARCHAR(1024)            -- Extended description of the product type
102   );
103   GO
104
105   -- Series classifier for products. Products may be part of multiple series. Series may also include brand labels.
106   CREATE TABLE series
107   (
108       series_id          INT IDENTITY PRIMARY KEY,        -- DB identifier of series
109       manufacturer_id    INT NOT NULL FOREIGN KEY REFERENCES manufacturer(manufacturer_id),
110                                                    -- Manufaturer owner of series
111       series_name        NVARCHAR(128) NOT NULL,          -- Human name of series
```

```sql
112      series_description  NVARCHAR(1024),              -- Extended description of series
113
114      CONSTRAINT u1_series UNIQUE (series_name, manufacturer_id)
115                                                   -- Each series name must be unique to its manufacturer
116  );
117  GO
118
119  -- Primary listing of products
120  CREATE TABLE product
121  (
122      product_id                INT IDENTITY PRIMARY KEY,
123                                                   -- DB identifier of product record
124      product_name              NVARCHAR(128) NOT NULL, -- Human identifer or product record
125      product_type_id           INT FOREIGN KEY REFERENCES product_type(product_type_id),
126                                                   -- Type of product
127      product_description       NVARCHAR(1024),        -- Extended description and notes about product
128      year_of_release           INT,                   -- Year product was released for sale by its manufacturer
129      purchase_price            DECIMAL(19,4),         -- Price the owner paid for the product
130      purchased_from_retailer_id  INT FOREIGN KEY REFERENCES retailer(retailer_id),
131                                                   -- Retailer from whom the owner purchased the product
132      manufacturer_id           INT FOREIGN KEY REFERENCES manufacturer(manufacturer_id),
133                                                   -- Product's manufacturer
134      exclusive_to_retailer_id   INT FOREIGN KEY REFERENCES retailer(retailer_id),
135                                                   -- Retailer that initially had exclusive sales rights to the product
136
137      CONSTRAINT u1_product UNIQUE (product_name, manufacturer_id)
138                                                   -- Allow only one unique product name from a manufacturer.
139                                                   -- If a manufacturer produces two products with the same name we
140                                                   --  require that the product name have something unique to distinguish it
141                                                   -- in the data record. If we have duplicates (which is the case for my
142                                                   -- collection) the we will have one product with multiple figures.
143                                                   -- The figures' descriptions will indicate that they are duplicates.
144  );
145  GO
146
147  -- Relationship(s) of products to specific series.
148  CREATE TABLE product_series
```

```sql
149  (
150      product_series_id    INT IDENTITY PRIMARY KEY,        -- DB identifier of record
151      product_id           INT FOREIGN KEY REFERENCES product(product_id),
152                                                   -- Product identifier
153      series_id            INT FOREIGN KEY REFERENCES series(series_id),
154                                                   -- Associated series identifier
155
156      -- Each product/seies combination must be unique.
157      CONSTRAINT u1_product_series UNIQUE (product_id, series_id)
158  );
159
160  -- Relationships of products to specific generes
161  CREATE TABLE product_genre
162  (
163      product_genre_id    INT IDENTITY PRIMARY KEY,        -- DB indentifier of record
164      product_id          INT FOREIGN KEY REFERENCES product(product_id),
165                                                   -- Product identifier
166      genre_id            INT FOREIGN KEY REFERENCES genre(genre_id),
167                                                   -- Associated genre identifier
168
169      -- Each product/genre combination must be unique.
170      CONSTRAINT u1_product_genre UNIQUE (product_id, genre_id)
171  );
172  GO
173
174  -- Individiual action figures within a product offering
175  CREATE TABLE action_figure
176  (
177    action_figure_id          INT IDENTITY PRIMARY KEY,    -- DB identifier of record
178    action_figure_description NVARCHAR(1024),             -- Detailed description (if needed) of action figure.
179    storage_location          VARCHAR(32) NOT NULL,       -- Location where the action figure is stored or displayed
180    likeness                  NVARCHAR(128),              -- Celebrity or historic figure  the action figure resembles
181    product_id                INT FOREIGN KEY REFERENCES product(product_id)
182                                                   -- Product to which the action figure belongs
183  );
184  GO
185
```

```sql
186  -- Relationship(s) of products to images
187  CREATE TABLE product_product_image
188  (
189      product_product_image_id  INT IDENTITY PRIMARY KEY,    -- DB indentifier of record
190      product_id                INT FOREIGN KEY REFERENCES product(product_id),
191                                                             -- Product identifier
192      product_image_id          INT FOREIGN KEY REFERENCES product_image(product_image_id),
193                                                             -- Associated image identifier
194
195      -- Each image may be associated with a single product at most one time.
196      CONSTRAINT u1_product_product_image UNIQUE (product_id, product_image_id)
197  );
198  GO
```