

SQLAlchemy/database.py (Page 1 of 1)

```
1: #!/usr/bin/env python
2:
3: #-----
4: # database.py
5: # Author: Bob Dondero
6: #-----
7:
8: import sqlalchemy.ext.declarative
9: import sqlalchemy
10:
11: Base = sqlalchemy.ext.declarative.declarative_base()
12:
13: class Book (Base):
14:     __tablename__ = 'books'
15:     isbn = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
16:     title = sqlalchemy.Column(sqlalchemy.String)
17:     quantity = sqlalchemy.Column(sqlalchemy.Integer)
18:
19: class Author (Base):
20:     __tablename__ = 'authors'
21:     isbn = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
22:     author = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
23:
24: class Order (Base):
25:     __tablename__ = 'orders'
26:     isbn = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
27:     custid = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
28:     quantity = sqlalchemy.Column(sqlalchemy.Integer)
29:
30: class Customer (Base):
31:     __tablename__ = 'customers'
32:     custid = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
33:     custname = sqlalchemy.Column(sqlalchemy.String)
34:     street = sqlalchemy.Column(sqlalchemy.String)
35:     zipcode = sqlalchemy.Column(sqlalchemy.String)
36:
37: class Zipcode (Base):
38:     __tablename__ = 'zipcodes'
39:     zipcode = sqlalchemy.Column(sqlalchemy.String, primary_key=True)
40:     city = sqlalchemy.Column(sqlalchemy.String)
41:     state = sqlalchemy.Column(sqlalchemy.String)
```

blank (Page 1 of 1)

```
1: This page is intentionally blank.
```

SQLAlchemy/create.py (Page 1 of 2)

```

1: #!/usr/bin/env python
2: #-----
3: # create.py
4: # Author: Bob Dondero
5: #-----
6: #
7:
8: import os
9: import sys
10: import sqlite3
11: import sqlalchemy
12: import sqlalchemy.orm
13: import dotenv
14: import database
15:
16: dotenv.load_dotenv()
17: DATABASE_URL = os.environ['DATABASE_URL']
18:
19: #-----
20:
21: def main():
22:
23:     if len(sys.argv) != 1:
24:         print('Usage: python ' + sys.argv[0], file=sys.stderr)
25:         sys.exit(1)
26:
27:     try:
28:         engine = sqlalchemy.create_engine(DATABASE_URL)
29:
30:         database.Base.metadata.drop_all(engine)
31:         database.Base.metadata.create_all(engine)
32:
33:         with sqlalchemy.orm.Session(engine) as session:
34:
35:             #-----
36:
37:             book = database.Book(isbn=123,
38:                                 title='The Practice of Programming', quantity=500)
39:             session.add(book)
40:             book = database.Book(isbn=234,
41:                                 title='The C Programming Language', quantity=800)
42:             session.add(book)
43:             book = database.Book(isbn=345,
44:                                 title='Algorithms in C', quantity=650)
45:             session.add(book)
46:             session.commit()
47:
48:             #-----
49:
50:             author = database.Author(isbn=123, author='Kernighan')
51:             session.add(author)
52:             author = database.Author(isbn=123, author='Pike')
53:             session.add(author)
54:             author = database.Author(isbn=234, author='Kernighan')
55:             session.add(author)
56:             author = database.Author(isbn=234, author='Ritchie')
57:             session.add(author)
58:             author = database.Author(isbn=345, author='Sedgewick')
59:             session.add(author)
60:             session.commit()
61:
62:             #-----
63:
64:             customer = database.Customer(custid='111',
65:                                         custname='Princeton', street='114 Nassau St',

```

SQLAlchemy/create.py (Page 2 of 2)

```

66:                                     zipcode='08540')
67:             session.add(customer)
68:             customer = database.Customer(custid='222',
69:                                         custname='Harvard', street='1256 Mass Ave',
70:                                         zipcode='02138')
71:             session.add(customer)
72:             customer = database.Customer(custid='333',
73:                                         custname='MIT', street='292 Main St',
74:                                         zipcode='02142')
75:             session.add(customer)
76:             session.commit()
77:
78:             #-----
79:
80:             zipcode = database.Zipcode(zipcode='08540',
81:                                       city='Princeton', state='NJ')
82:             session.add(zipcode)
83:             zipcode = database.Zipcode(zipcode='02138',
84:                                       city='Cambridge', state='MA')
85:             session.add(zipcode)
86:             zipcode = database.Zipcode(zipcode='02142',
87:                                       city='Cambridge', state='MA')
88:             session.add(zipcode)
89:             session.commit()
90:
91:             #-----
92:
93:             order = database.Order(isbn='123', custid='222',
94:                                   quantity=20)
95:             session.add(order)
96:             order = database.Order(isbn='345', custid='222',
97:                                   quantity=100)
98:             session.add(order)
99:             order = database.Order(isbn='123', custid='111',
100:                                   quantity=30)
101:             session.add(order)
102:             session.commit()
103:
104:             engine.dispose()
105:
106:         except Exception as ex:
107:             print(ex, file=sys.stderr)
108:             sys.exit(1)
109:
110: #-----
111:
112: if __name__ == '__main__':
113:     main()

```

SQLAlchemy/display.py (Page 1 of 2)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # display.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import sqlite3
11: import sqlalchemy
12: import sqlalchemy.orm
13: import dotenv
14: import database
15:
16: dotenv.load_dotenv()
17: DATABASE_URL = os.environ['DATABASE_URL']
18:
19: #-----
20:
21: def main():
22:
23:     if len(sys.argv) != 1:
24:         print('Usage: python ' + sys.argv[0], file=sys.stderr)
25:         sys.exit(1)
26:
27:     try:
28:         engine = sqlalchemy.create_engine(DATABASE_URL)
29:
30:         with sqlalchemy.orm.Session(engine) as session:
31:
32:             print('-----')
33:             print('books')
34:             print('-----')
35:             query = session.query(database.Book)
36:             table = query.all()
37:             for row in table:
38:                 print(row.isbn, row.title, row.quantity)
39:
40:             print('-----')
41:             print('authors')
42:             print('-----')
43:             query = session.query(database.Author)
44:             table = query.all()
45:             for row in table:
46:                 print(row.isbn, row.author)
47:
48:             print('-----')
49:             print('customers')
50:             print('-----')
51:             query = session.query(database.Customer)
52:             table = query.all()
53:             for row in table:
54:                 print(row.custid, row.custname, row.street, row.zipcode)
55:
56:             print('-----')
57:             print('zipcodes')
58:             print('-----')
59:             query = session.query(database.Zipcode)
60:             table = query.all()
61:             for row in table:
62:                 print(row.zipcode, row.city, row.state)
63:
64:             print('-----')
65:             print('orders')

```

SQLAlchemy/display.py (Page 2 of 2)

```

66:             print('-----')
67:             query = session.query(database.Order)
68:             table = query.all()
69:             for row in table:
70:                 print(row.isbn, row.custid, row.quantity)
71:
72:             engine.dispose()
73:
74:         except Exception as ex:
75:             print(ex, file=sys.stderr)
76:             sys.exit(1)
77:
78: #-----
79:
80: if __name__ == '__main__':
81:     main()

```

SQLAlchemy/authorsearch.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # authorsearch.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import sqlite3
11: import sqlalchemy
12: import sqlalchemy.orm
13: import dotenv
14: import database
15:
16: dotenv.load_dotenv()
17: DATABASE_URL = os.environ['DATABASE_URL']
18:
19: #-----
20: #
21: def main():
22:
23:     if len(sys.argv) != 2:
24:         print('Usage: python ' + sys.argv[0] + ' author',
25:               file=sys.stderr)
26:         sys.exit(1)
27:
28:     author = sys.argv[1]
29:
30:     try:
31:         engine = sqlalchemy.create_engine(DATABASE_URL)
32:
33:         try:
34:             with sqlalchemy.orm.Session(engine) as session:
35:
36:                 query = (session.query(database.Book.isbn,
37:                                       database.Book.title, database.Book.quantity)
38:                          .filter(database.Book.isbn == database.Author.isbn)
39:                          .filter(database.Author.author == author))
40:
41:                 table = query.all()
42:                 for row in table:
43:                     print('ISBN:', row.isbn)
44:                     print('Title:', row.title)
45:                     print('Quantity:', row.quantity)
46:                     print()
47:             finally:
48:                 engine.dispose()
49:
50:         except Exception as ex:
51:             print(ex, file=sys.stderr)
52:             sys.exit(1)
53:
54: #-----
55:
56: if __name__ == '__main__':
57:     main()

```

SQLAlchemy/order.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # order.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import sqlite3
11: import sqlalchemy
12: import sqlalchemy.orm
13: import dotenv
14: import database
15:
16: dotenv.load_dotenv()
17: DATABASE_URL = os.environ['DATABASE_URL']
18:
19: #-----
20: #
21: def main():
22:
23:     if len(sys.argv) != 3:
24:         print('Usage: python ' + sys.argv[0] + ' isbn custid',
25:               file=sys.stderr)
26:         sys.exit(1)
27:
28:     isbn = sys.argv[1]
29:     custid = sys.argv[2]
30:
31:     try:
32:         engine = sqlalchemy.create_engine(DATABASE_URL)
33:
34:         with sqlalchemy.orm.Session(engine) as session:
35:
36:             query = (session.query(database.Order)
37:                     .filter(database.Order.isbn == isbn)
38:                     .filter(database.Order.custid == custid))
39:
40:             row = query.one()
41:             row.quantity += 1
42:
43:             session.commit()
44:
45:             engine.dispose()
46:
47:         except Exception as ex:
48:             print(ex, file=sys.stderr)
49:             sys.exit(1)
50:
51: #-----
52:
53: if __name__ == '__main__':
54:     main()

```

SQLAlchemy/purchase.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # purchase.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import sqlite3
11: import sqlalchemy
12: import sqlalchemy.orm
13: import dotenv
14: import database
15:
16: dotenv.load_dotenv()
17: DATABASE_URL = os.environ['DATABASE_URL']
18:
19: #-----
20:
21: def main():
22:
23:     if len(sys.argv) != 3:
24:         print('Usage: python ' + sys.argv[0] + ' isbn custid',
25:               file=sys.stderr)
26:         sys.exit(1)
27:
28:     isbn = int(sys.argv[1])
29:     custid = int(sys.argv[2])
30:
31:     try:
32:         engine = sqlalchemy.create_engine(DATABASE_URL)
33:
34:         with sqlalchemy.orm.Session(engine) as session:
35:
36:             query = (session.query(database.Order)
37:                     .filter(database.Order.isbn == isbn)
38:                     .filter(database.Order.custid == custid))
39:             row = query.one()
40:             row.quantity += 1
41:
42:             query = (session.query(database.Book)
43:                     .filter(database.Book.isbn == isbn))
44:             row = query.one()
45:             row.quantity -= 1
46:
47:             session.commit()
48:             print('Transaction committed.')
49:
50:         engine.dispose()
51:
52:     except Exception as ex:
53:         print(ex, file=sys.stderr)
54:         sys.exit(1)
55:
56: #-----
57:
58: if __name__ == '__main__':
59:     main()

```

SQLAlchemy/recovery.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # recovery.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import random
11: import sqlite3
12: import sqlalchemy
13: import sqlalchemy.orm
14: import dotenv
15: import database
16:
17: dotenv.load_dotenv()
18: DATABASE_URL = os.environ['DATABASE_URL']
19:
20: #-----
21:
22: def main():
23:
24:     if len(sys.argv) != 1:
25:         print('Usage: python ' + sys.argv[0], file=sys.stderr)
26:         sys.exit(1)
27:
28:     try:
29:         engine = sqlalchemy.create_engine(DATABASE_URL)
30:         with sqlalchemy.orm.Session(engine) as session:
31:
32:             for i in range(20):
33:
34:                 query = (session.query(database.Order)
35:                         .filter(database.Order.isbn == '123')
36:                         .filter(database.Order.custid == '222'))
37:                 row = query.one()
38:                 row.quantity += 1
39:
40:                 # Simulate a HW/SW failure occurring randomly,
41:                 # on average every 5th time through the loop.
42:                 if random.randrange(5) == 0:
43:                     print('Simulated failure with i = %d' % i)
44:                     session.rollback()
45:                     print('Transaction rolled back.')
46:                     continue
47:
48:                 query = (session.query(database.Book)
49:                         .filter(database.Book.isbn == '123'))
50:                 row = query.one()
51:                 row.quantity -= 1
52:
53:                 session.commit()
54:                 print('Transaction %d committed.' % i)
55:
56:             engine.dispose()
57:
58:     except Exception as ex:
59:         print(ex, file=sys.stderr)
60:         sys.exit(1)
61:
62: #-----
63:
64: if __name__ == '__main__':
65:     main()

```