[[One To Many](http://docs.sqlalchemy.org/en/latest/orm/basic_relationships.html#one-to-many)]

class Article(Base):

\_\_tablename\_\_ = 'articles'

id = Column(Integer, primary\_key=True)

comments = relationship("Comment")

class Comment(Base):

\_\_tablename\_\_ = 'comments'

id = Column(Integer, primary\_key=True)

article\_id = Column(Integer, ForeignKey('articles.id'))

[ [Many To One](http://docs.sqlalchemy.org/en/latest/orm/basic_relationships.html#many-to-one) ]

class Tire(Base):

\_\_tablename\_\_ = 'tires'

id = Column(Integer, primary\_key=True)

car\_id = Column(Integer, ForeignKey('cars.id'))

car = relationship("Car")

class Car(Base):

\_\_tablename\_\_ = 'cars'

id = Column(Integer, primary\_key=True)

[ [One To One](http://docs.sqlalchemy.org/en/latest/orm/basic_relationships.html#one-to-one) ]

class Person(Base):

\_\_tablename\_\_ = 'people'

id = Column(Integer, primary\_key=True)

mobile\_phone = relationship("MobilePhone", uselist=False, back\_populates="person")

class MobilePhone(Base):

\_\_tablename\_\_ = 'mobile\_phones'

id = Column(Integer, primary\_key=True)

person\_id = Column(Integer, ForeignKey('people.id'))

person = relationship("Person", back\_populates="mobile\_phone")

[ [Many To Many](http://docs.sqlalchemy.org/en/latest/orm/basic_relationships.html#many-to-many) ]

students\_classes\_association = Table('students\_classes', Base.metadata,

Column('student\_id', Integer, ForeignKey('students.id')),

Column('class\_id', Integer, ForeignKey('classes.id'))

)

class Student(Base):

\_\_tablename\_\_ = 'students'

id = Column(Integer, primary\_key=True)

classes = relationship("Class", secondary=students\_classes\_association)

class Class(Base):

\_\_tablename\_\_ = 'classes'

id = Column(Integer, primary\_key=True)