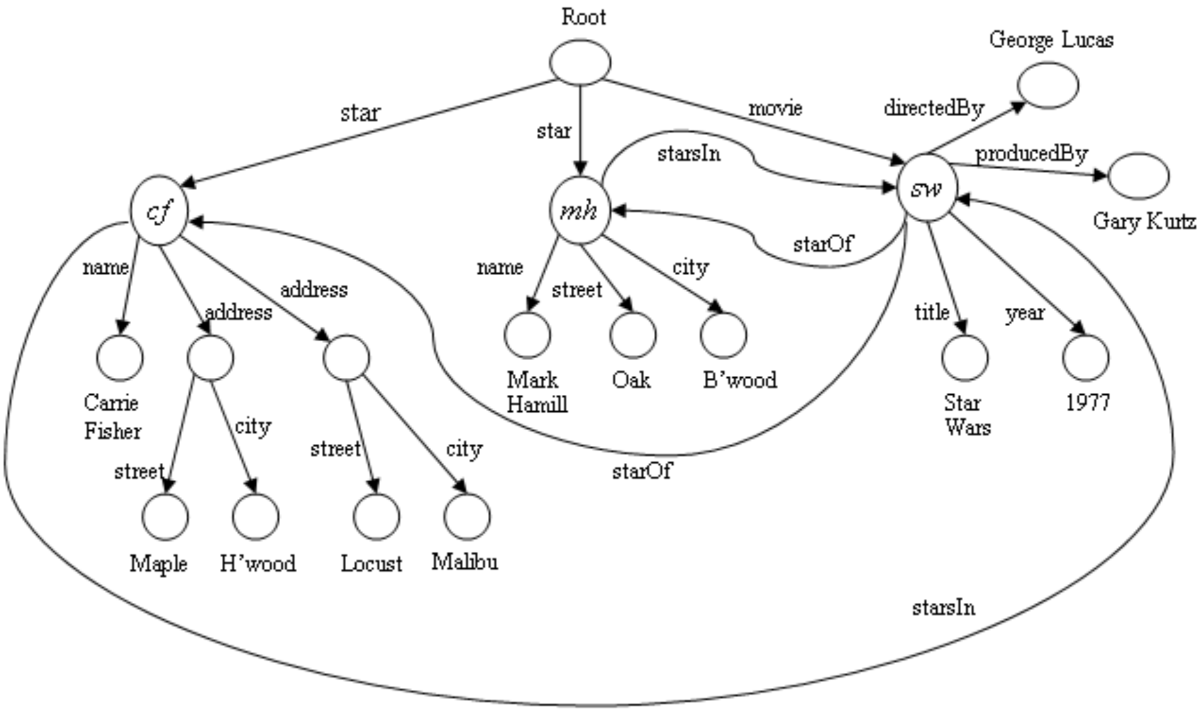
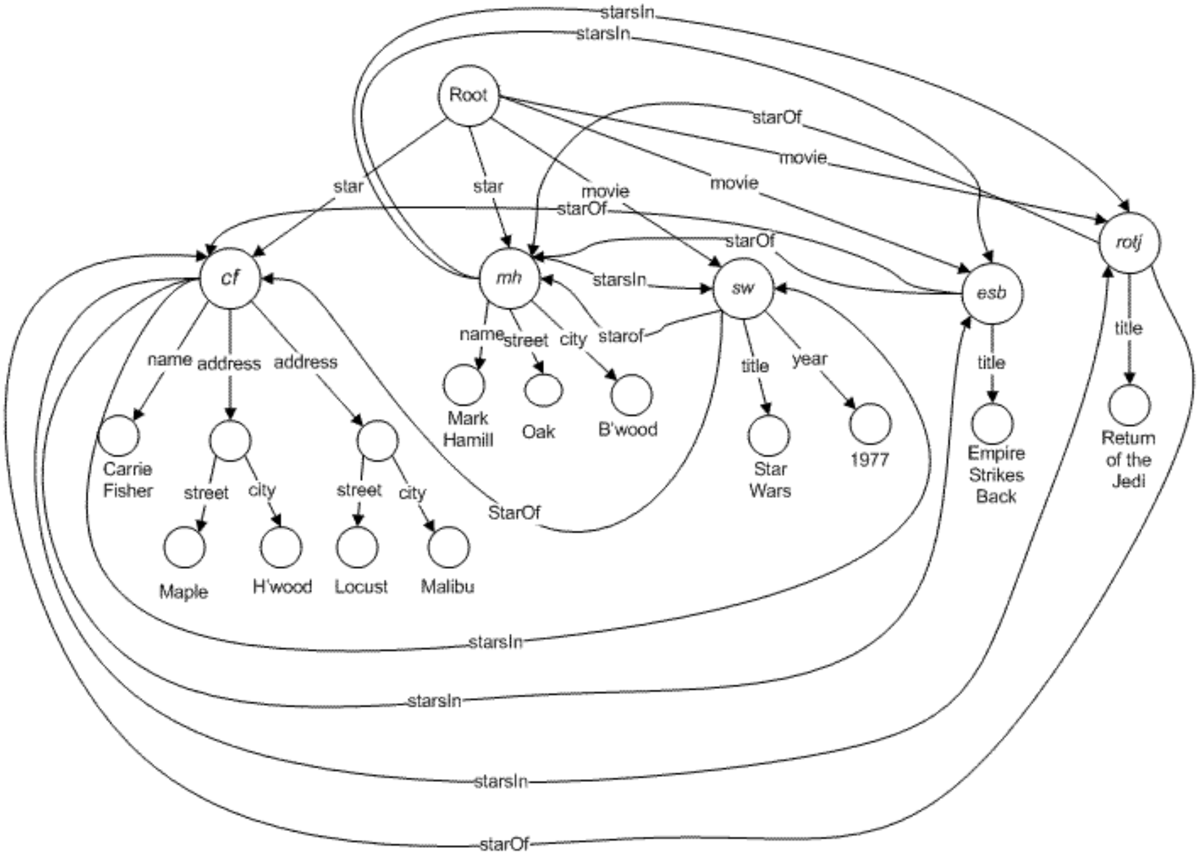
## Section 1

### Exercise 11.1.1

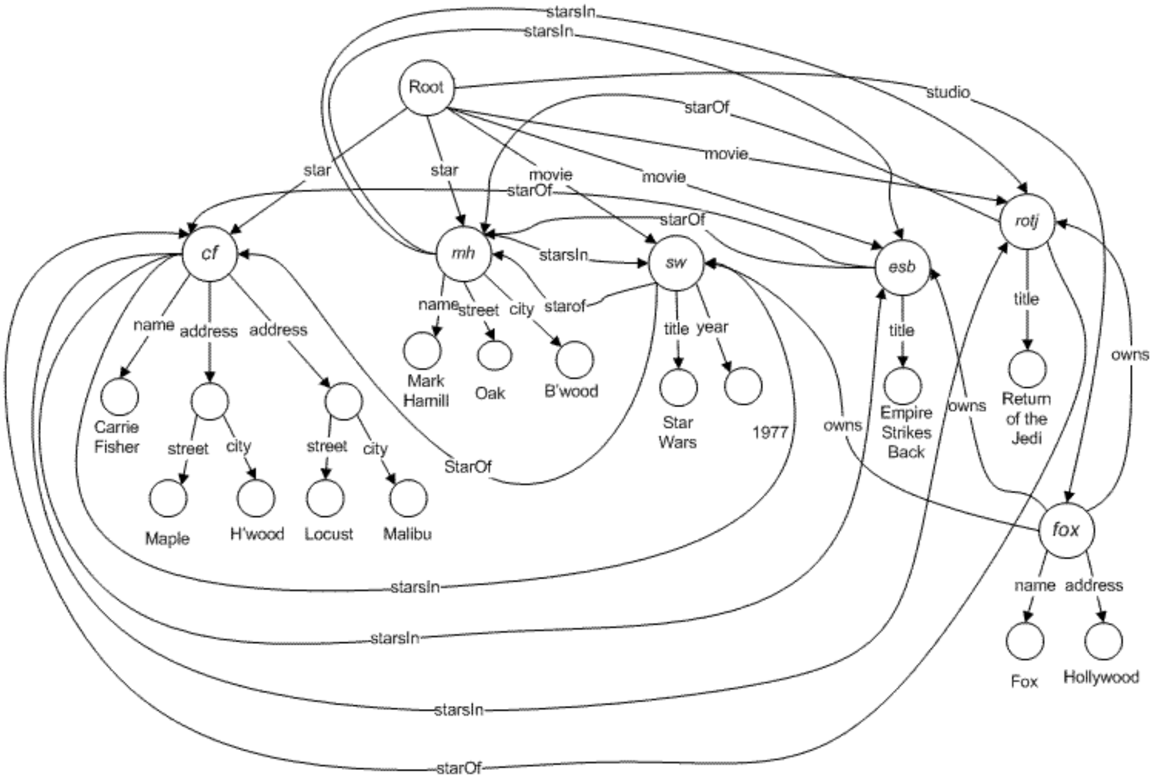
a)



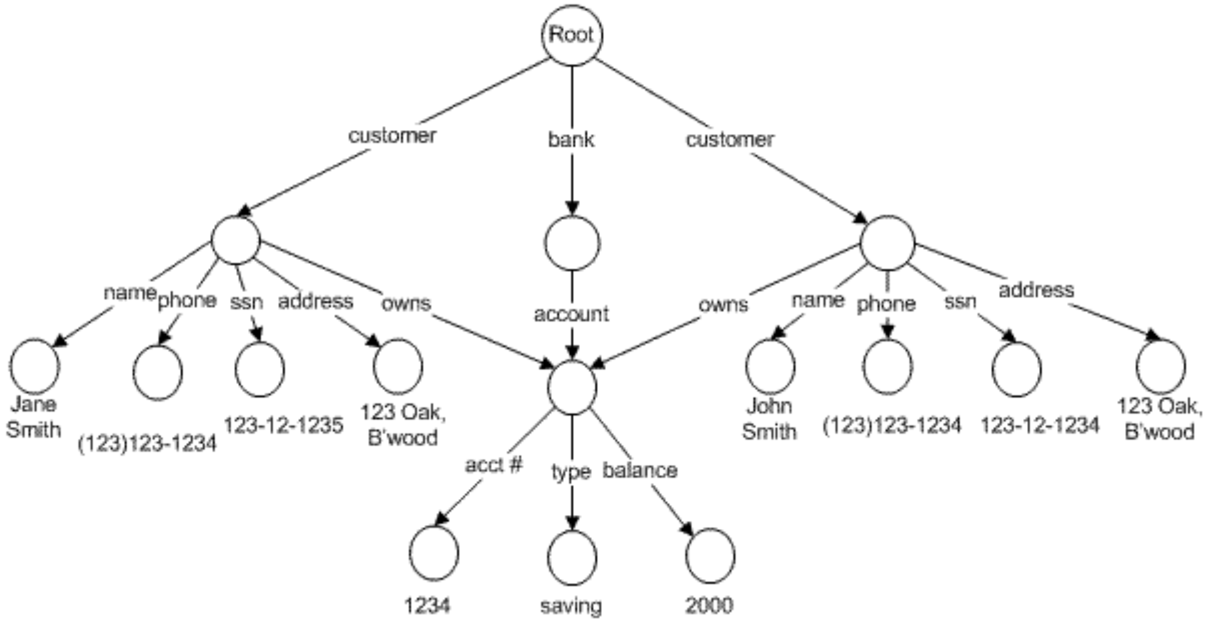
b)


starsIn
starsIn
Root
starOf
movie
star
star
movie
movie
starOf
starOf
cf
mh
starsIn
sw
esb
rotj
name
address
address
name
street
city
starof
title
year
title
title
Carrie Fisher
Mark Hamill
Oak
B'wood
Star Wars
1977
Empire Strikes Back
Return of the Jedi
street
city
street
city
StarOf
Maple
H'wood
Locust
Malibu
starsIn
starsIn
starsIn
starOf

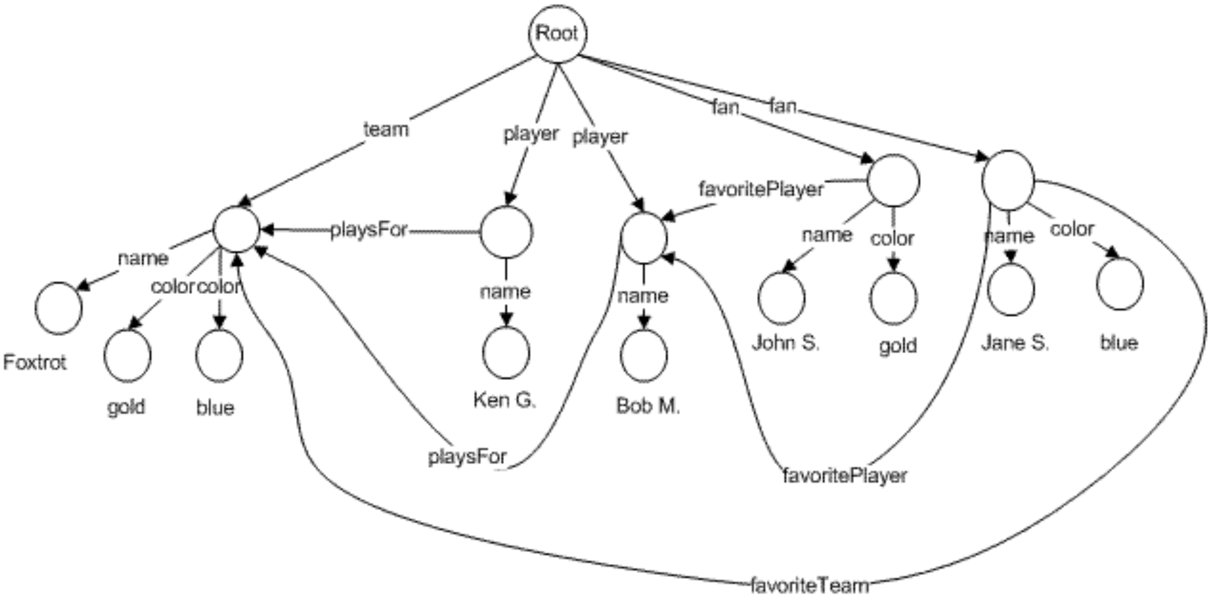
c)


starsIn
starsIn
Root
studio
starOf
movie
star
star
movie
movie
starOf
starOf
rotj
cf
mh
starsIn
sw
esb
starof
name
address
address
name
street
city
title
year
title
title
owns
Mark
Hamill
Oak
B'wood
Star
Wars
1977
Empire
Strikes
Back
Return
of the
Jedi
Carrie
Fisher
street
city
street
city
StarOf
owns
owns
Maple
H'wood
Locust
Malibu
fox
name
address
Fox
Hollywood
starsIn
starsIn
starsIn
starOf

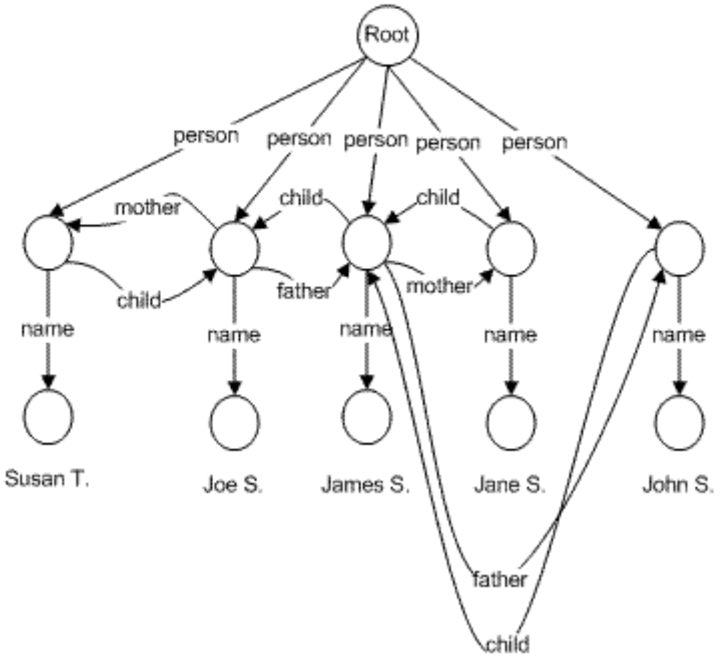
## Exercise 11.1.2



## Exercise 11.1.3



## Exercise 11.1.4



## Exercise 11.1.5

In the semistructured model, nodes represent data elements, i.e., entities rather than entity sets. In the UML model, nodes of all types represent object sets, and the data is not represented at all.

## Section 2

## Exercise 11.2.1

```
<? xml version = "1.0" encoding = "utf-8" standalone = "yes" ?>
<StarMovieData>
        <Star starID = "cf" starredIn = "sw">
                <Name>Carrie Fisher</Name>
                <Address>
                        <Street>123 Maple St.</Street>
                        <City>Hollywood</City>
                </Address>
                <Address>
                        <Street>5 Locust Ln.</Stree>
                        <City>Malibu</City>
                </Address>
        </Star>
        <Star starID = "mh" starredIn = "sw">
                <Name>Mark Hamill</Name>
                <Address>
                <Street>456 Oak Rd.</Street>
                <City>Brentwood</City>
            </Address>
        </Star>
        <Movie movieID = "sw" starsOf = "cf mh">
            <Title>Star Wars</Title>
            <Year>1977</Year>
        </Movie>
</StarMovieData>
```

**Exercise 11.2.2**

| Title | Year | Length | Genre |
|---|---|---|---|
| Gone with the Wind | 1939 | 231 | Drama |
| Star Wars | 1977 | 124 | SciFi |
| Wayne’s World | 1992 | 95 | Comedy |

Figure 2.3: The Relation Movies

The relation Movies in XML:

```
<Movies>
        <Movie>
                <Title>Gone with the Wing</title>
                <Year>1939</year>
                <Length>231</length>
                <Genre>Drama</genre>
        </Movie>
        <Movie>
                <Title>Star Wars</Title>
                <Year>1977</year>
                <Length>124</length>
                <Genre>SciFi</genre>
        </Movie>
        <Movie>
                <Title>Wayne’s World</Title>
                <Year>1992</year>
                <Length>95</length>
                <Genre>Comedy</genre>
        </Movie>
</Movies>
```

**Exercise 11.2.3**

An empty element can be represented as a SubElement whose childID value is null (or a predefined ID that does not map to any element).

**Exercise 11.2.4**

DocRoot(docID, rootElementID)
Element(elementID, position)
ElementAttribute(elementID, name, value)
ElementValue(elementID, value)
SubElementOf(parentID, childID)

**Section 3**

Exercise 11.3.1

```
<?xml version = "1.0" encoding = "utf-8" standalone = "yes" ?>
<StarMovieData>
        <Star starID = "cf" starredIn = "sw tesb rotj hahs">
                <Name>Carrie Fisher</Name>
                <Address>
                        <Street>123 Maple St.</Street>
                        <City>Hollywood</City>
                </Address>
                <Address>
                        <Street>5 Locust Ln.</Street>
                        <City>Malibu</City>
                </Address>
        </Star>
        <Star starID = "mh" starredIn = "sw tesb rotj">
                <Name>Mark Hamill</Name>
                <Address>
                        <Street>456 Oak Rd.</Street>
                        <City>Brentwood</City>
                </Address>
        </Star>
        <Star starID = "hf" starredIn = "sw tesb rotj fw">
                <Name>Harrison Ford</Name>
        </Star>
        <Star starid = "md" starredIn = "tbi">
                <Name>Matt Damon</Name>
        </Star>
        <Movie movieID = "sw" starsOf = "cf mh hf">
                <Title>Star Wars</Title>
                <Year>1977</Year>
        </Movie>
        <Movie movieID = "tesb" starsOf = "cf mh hf">
                <Title>The Empire Strikes Back</Title>
```

```
            <Year>1980</Year>
        </Movie>
        <Movie movieID = "rotj" starsOf = "cf mh hf">
            <Title>Return of the Jedi</Title>
            <Year>1983</Year>
        </Movie>
        <Movie movieID = "fw" starsOf = "hf">
            <Title>Firewall</Title>
            <Year>2006</Year>
        </Movie>
        <Movie movieID = "hahs" starsOf = "cf">
            <Title>Hannah and Her Sisters</Title>
            <Year>1985</Year>
        </Movie>
        <Movie movieID = "tbi" starsOf = "md">
            <Title>The Bourne Identity</Title>
            <Year>2002</Year>
        </Movie>
</StarMovieData>
```

**Exercise 11.3.2**

```
<!DOCTYPE Bank [
         <!ELEMENT BANK (CUSTOMER*, ACCOUNT*)>
         <!ELEMENT CUSTOMER (NAME, ADDRESS, PHONE, SSNO)>
             <!ATTLIST CUSTOMER
                 custId ID
                 owns IDREFS>
         <!ELEMENT NAME (#PCDATA)>
         <!ELEMENT ADDRESS (#PCDATA)>
         <!ELEMENT PHONE (#PCDATA)>
         <!ELEMENT SSNO (#PCDATA)>
         <!ELEMENT ACCOUNT (NUMBER, TYPE, BALANCE)>
             <!ATTLIST ACCOUNT
                 acctId ID
                 ownedBy IDREFS>
         <!ELEMENT NUMBER (#PCDATA)>
         <!ELEMENT TYPE (#PCDATA)>
         <!ELEMENT BALANCE (#PCDATA)>
     ]>
```

**Exercise 11.3.3**

```
<!DOCTYPE Sport [
    <!ELEMENT SPORT(TEAM*, FAN*)>
    <!ELEMENT TEAM(NAME, PLAYER+, COLOR+)>
        <!ATTLIST TEAM
            captain IDREF #REQUIRED
            teamID ID #REQUIRED>
    <!ELEMENT PLAYER(NAME)>
        <ATTLIST PLAYER
            playerID ID #REQUIRED>
    <!ELEMENT FAN(NAME, COLOR*)>
        <ATTLIST FAN
            favoriteTeams IDREFS
            favoritePlayers IDREFS>
    <!ELEMENT COLOR (#PCDATA)>
    <!ELEMENT NAME (#PCDATA)>
]>
```

**Exercise 11.3.4**

```
<!DOCTYPE Genealogy [
    <!ELEMENT GENEALOGY(PERSON*)>
    <!ELEMENT PERSON EMPTY>
        <ATTLIST PERSON
            name CDATA #REQUIRED
            personID ID #REQUIRED
            motherID IDREF
            fatherID IDREF
            childrenID IDREFS>
]>
```

**Exercise 11.3.5**

Relation schema: RELATIONNAME(attribute list)

```
<!DOCTYPE RELATIONNAME [
    <!ELEMENT RELATIONNAME(ROW*)>
    <!ELEMENT ROW(attribute list)>
    for each attribute T in the attribute list
        <!ELEMENT T (#PCDATA)>
]
```

note: the singular form of the relation name should used in place of ROW

**Section 4**

## Exercise 11.4.1

```
<Movies>
    <Movie>
          <Title>Star Wars</Title>
          <Year>1977</Year>
    </Movie>
</Movies>

<Movies>
    <Movie>
          <Title>Star Wars</Title>
          <Year>'77</Year> <!-- not an integer -->
    </Movie>
</Movies>
```

## Exercise 11.4.2

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexType name="Movies">
        <xs:sequence>
              <xs:element name="Movie" minOccurs="0" maxOccurs="unbounded">
                  <xs:complexType>
                        <xs:sequence>
                            <xs:element name="Title" type="xs:string" />
                            <xs:element name="Year" type="xs:integer" />
                        </xs:sequence>
                    </xs:complexType>
              </xs:element>
          </xs:sequence>
    </xs:complexType>
</xs:schema>
```

**Exercise 11.4.3**

```
<!DOCTYPE MOVIES [
        <!ELEMENT Title (#PCDATA)>
        <!ELEMENT Year (#PCDATA)>
        <!ELEMENT Genre (#PCDATA)>
        <!ELEMENT Movies (Movie*)>
        <!ELEMENT Movie (Title, Year, Genre?)>
]>

<!DOCTYPE STARS [
        <!ELEMENT Stars (Star+)>
        <!ELEMENT Star (Name, Address, StarredIn*)>
        <!ELEMENT Name (#PCDATA)>
        <!ELEMENT Address (#PCDATA)>
        <!ELEMENT StarredIn EMPTY>
        <!ATTLIST StarredIn
                title CDATA #IMPLIED
                year CDATA #IMPLIED>
]>
```

Note: DTD is not as granular as XML schema; some restrictions in XML schema cannot be enforced in DTD without modifying the structures of the XML it defines.