Practice Midterm Exam

You will have 90 minutes for this exam, although you should not need that much. This exam is open-book and open-note, but you may not use the computer for anything other than accessing DyKnow notes or PDF files from Moodle. Please take some time to check your work. If you need extra space, write on the back. There are a total of 30 points on this exam.

For several questions on this exam, you will need the following database:

Employee

ssn	ename	age	salary
314159265	Ray Diaz	35	89000
271828182	Nathan Logg	42	78000
161803398	Phyllis Bonacci	34	55000
030102999	Bryan Decimali	32	65000
141421356	Hiram Pottanoose	50	100000

Works

essn	did	hours
314159265	1	20
314159265	2	20
271828182	1	10
271828182	2	20
271828182	3	5
271828182	4	5
161803398	2	10
161803398	3	30
030102999	2	20
030102999	3	20
030102999	4	20
141421356	4	40

Department

deptid	dname	mgrssn	budget
1	Hardware	314159265	531000
2	Firmware	314159265	420000
3	Software	161803398	678000
4	Sleepwear	141421356	88000

The Works table records the hours an employee spends working for a department each week; the mgrssn field identifies the manager of a department. Here is an SQL specification of these tables:

```
CREATE TABLE Employee (
  ssn INT not null,
  ename VARCHAR(20) not null,
  age INT not null,
  salary INT not null,
  PRIMARY KEY (ssn)
);

CREATE TABLE Department (
  deptid INT not null,
  dname VARCHAR(20) not null,
  mgrssn INT not null,
  budget INT not null,
  PRIMARY KEY (deptid),
  FOREIGN KEY (mgrssn) REFERENCES Employee
);

CREATE TABLE Works (
  essn INT not null,
  did INT not null,
  hours INT not null,
  FOREIGN KEY (essn) REFERENCES Employee,
  FOREIGN KEY (did) REFERENCES Department
);
```

1. (2 points) Draw a class diagram for the above database.

2. (3 points) List three different kinds of constraints expressed in the schema for this database.

3. (1 point) What would be a useful integrity constraint for this database?

4. (8 points) For each of the following SQL queries against the above database, show the resulting table.

   (a) ```
       SELECT ename, salary
       FROM Employee
       WHERE age < 40;
       ```

   (b) ```
       SELECT dname, ename, budget, salary
       FROM Employee, Dept
       WHERE ssn = mgrssn
       ORDER BY budget DESC;
       ```

   (c) ```
       SELECT dname, SUM(salary)
       FROM Employee, Dept, Works
       WHERE ssn = essn AND deptid = did AND hours >= 20
       GROUP BY deptid, dname;
       ```

5. (8 points) Write the following queries in SQL against the above database schema.

   (a) Retrieve the names and ages of all employees who work in both the Firmware department and the Software department.

   (b) Retrieve the names of all employees who are not managers.

   (c) Retrieve the name of the manager of the department with the largest budget.

6. (2 points) We saw that the relational algebra join operation can be implemented using a product followed by a select. How can you use the join operation to implement the product?

7. (3 points) Finish this JDBC code fragment which executes the query from problem 4(a) above and prints the result in a simple table.

```
Statement stmt = conn.createStatement();
String query = "SELECT ename, salary FROM Employee WHERE age < 40";
ResultSet rs = stmt.executeQuery(query);

System.out.format("%20s %10s", "Employee name", "Salary");
while (                         ) {




    System.out.format("%20s %10d",                ,              );
}

rs.close();
```

(The format string `"%20s %10d"` says to print a string 20 columns wide, a space, and an integer 10 columns wide.)

8. (3 points) Suppose the Employee table in the above database had an additional column identifying each employee's supervisor (by giving their ssn). Does this represent a redundant relationship? Why or why not? What might be a better way to store this information?