Timothy Rine
CSC 4402
HW4

①
```
CREATE TABLE Athletes
{
Name VARCHAR(255)
DOB CHAR(10),
PRIMARY KEY (Name, DOB)
};

CREATE TABLE Sports
{
Name VARCHAR(255) primary key,
isOlympic BIT,
FOREIGN KEY (Name) references (Name)
}
```

In these create table statements I made the foreign key 'name' reference the pk 'name' from the athlete table.

② R(ABCD)
$F = \{A \rightarrow BC, B \rightarrow D, A \rightarrow D\}$

(a) True. Because $B \rightarrow D$, we can also say that $D \rightarrow B$. There fore $A \rightarrow DC$, which is the same as $A \rightarrow CD$, is valid.

(b) True.
$A+ = ABCD$

(c) $F_c = \{$
$A \rightarrow BC$
$B \rightarrow D$
$\}$

Yes, this is redundant because $A \rightarrow D$ is already implied due to $A \rightarrow B$ and $B \rightarrow D$.

(3) (a) There are 3 candidate keys: AB, AC, AD

(b) Yes, R is in 3NF because FDs AB→C; AC→D have a ck on the left and the last FD has part of {AB} on the right. FD D→B.

(c) ABCD is decomposed into R1(<u>ACD</u>), with the keys AC, AD, and the FDs AC→D, AD→C, and R2(<u>D</u>B) with key D and FD D→B. Both R1 + R2 are in BCNF so decomposition ends.

(4) (a) I would use a hash index on id. The index is on the pk so <u>at most</u> one record can match. If a match is found then it will take 1 page from disk.

(b) I would use the unclustered B+-tree index on (age, rating). The leaves will have the data that we query for and it won't take pages from disk.