

**Washington State University**  
**School of Electrical Engineering and Computer Science**  
**CptS 451 – Introduction to Database Systems**

Dr. Sakire Arslan Ay

## **Homework-6**

Due Date: Saturday, June 27, 11:59pm
--------------------------------------

Name: \_\_\_\_\_

Student Number: \_\_\_\_\_

## Storage and Indexing

Consider the following schema:

Emp (eid, ename, addr, sal, age, yrs, deptid)

Dept (did, dname, floor, budget)

Primary keys are underlined.

Emp.deptid is a foreign key referencing to Dept.did.

Suppose you know that the following queries are the five most common queries in the workload for this university and all five are roughly equivalent in frequency and importance:

1. List the id, name, and address of employees who work in the department with a user-specified department name.
2. List the id and address of employees with a user-specified employee name.
3. List the overall average salary for employees.
4. List the average salary for employees of each age; that is, for each age in the database, list the age and the corresponding average salary.
5. List all the department information, ordered by department floor numbers.

These queries occur much more frequently than updates, so you should build whatever indexes you need to speed up these queries. However, you should not build any unnecessary indexes (or include any unnecessary attributes in an index), as updates will occur (and would be slowed down by unnecessary indexes). Given this information, decide which attributes should be indexed and whether each index should be a clustered index or an unclustered index. Assume that both B+ trees and hashed indexes are supported by the DBMS, and that both single- and multiple-attribute index keys are permitted.

For each index:

- identify the attributes you recommend indexing on (you can propose to create one or more new indexes for each query or you may suggest to re-use the indexes that are already created (proposed) for other queries. )
- indicate whether each index should be clustered or unclustered, and
- whether it should be a B+ tree or a hashed index.
- briefly describe how that index will be used to answer the query.

### Submission Instructions:

HW6 will be submitted on Blackboard. You may either type it or handwrite and scan it as pdf (only neat and readable handwriting please).