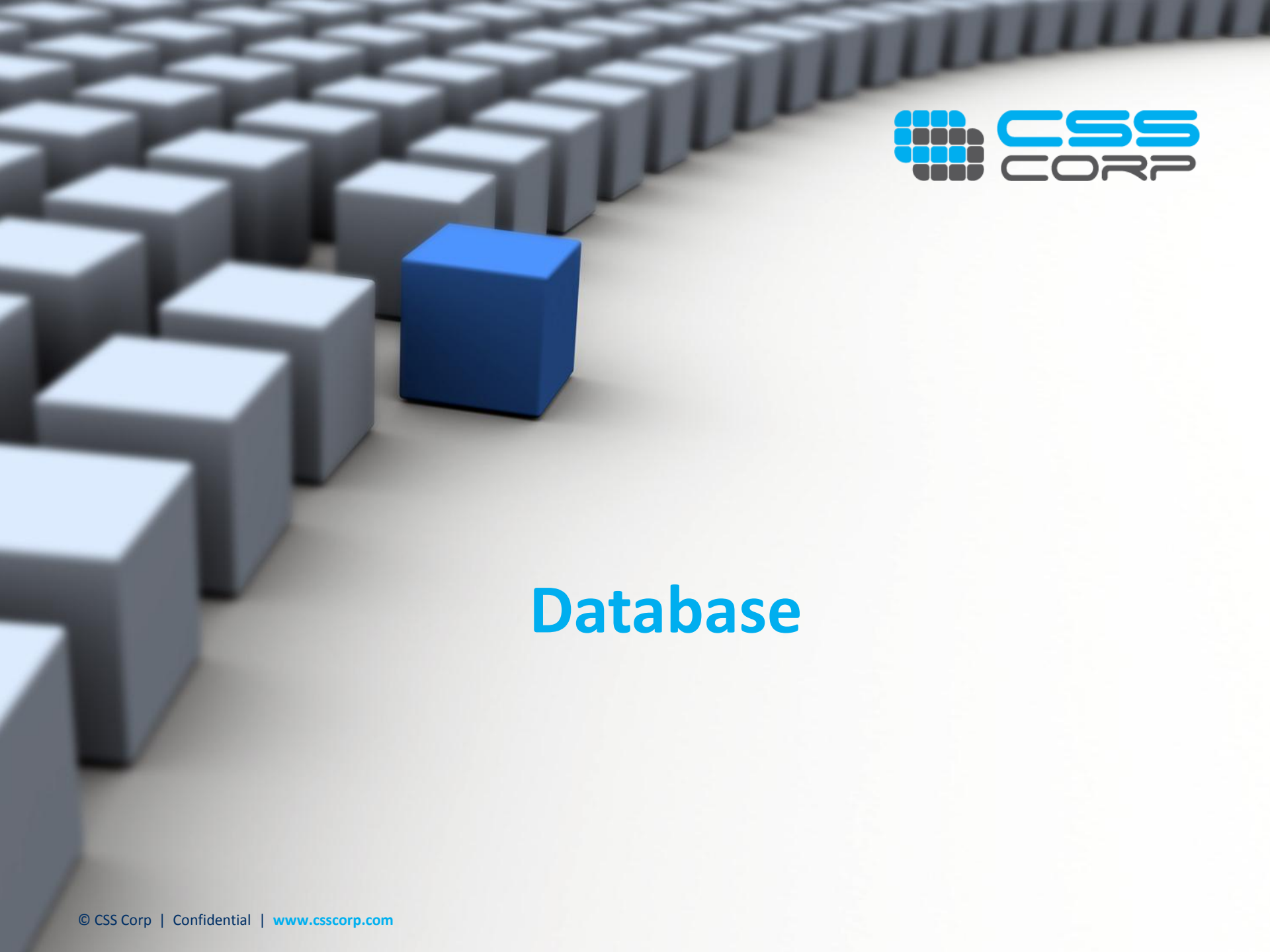
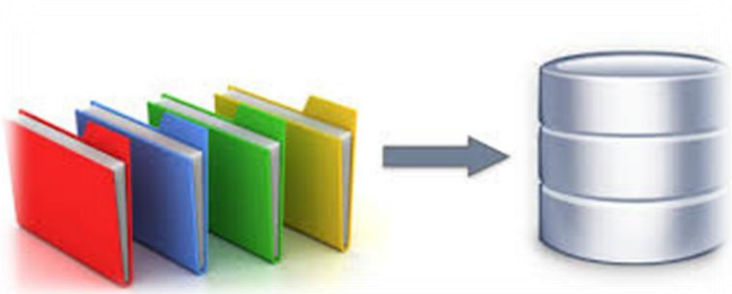
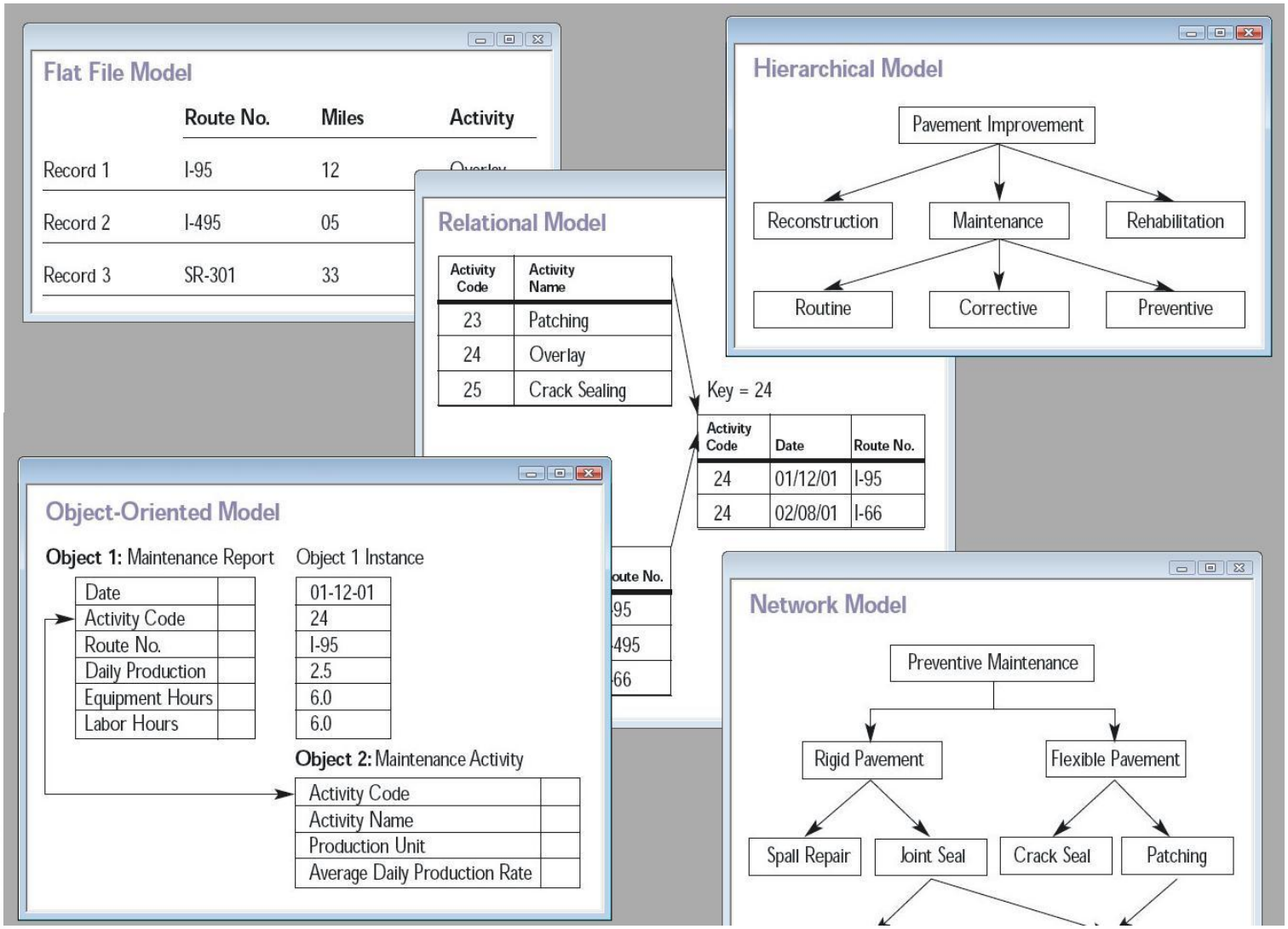


Database

What is Database???

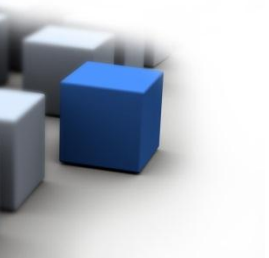


Modes of storing data

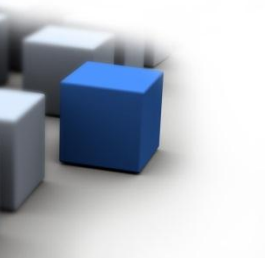


Modes of storing data

- Hierarchical Data model
  - Tree Structure
  - IBM’s Information management system, IMS
- Network Data Model
  - Pointers
  - Honeywell's Integrated Data Store, IDS
- Relational Data Model
  - Tables
  - MSSQL , Oracle , Sybase

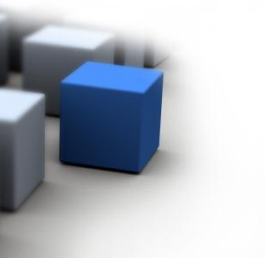


NORMALIZATION



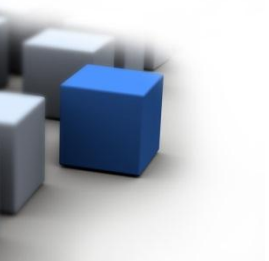
What is Normalization?

- Database designed based on Entity Relationship (ER) model may have some amount of inconsistency, ambiguity and redundancy.
- To resolve these issues some amount of refinement is required.
- This refinement process is called as **Normalization**.



Agenda

- what is the need of normalization?
- What are the problems we can face if we proceed without normalization?
- What are the advantages of normalization?



The need of Normalization

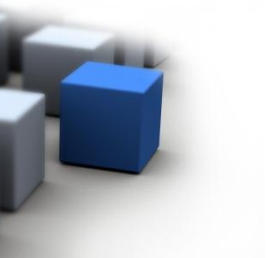
Student Details			Course Details			Result details		
1001	Ram	11/09/1986	M4	Basic Maths	7	11/11/2004	89	A
1002	Shyam	12/08/1987	M4	Basic Maths	7	11/11/2004	78	B
1001	Ram	23/06/1987	H6		4	11/11/2004	87	A
1003	Sita	16/07/1985	C3	Basic Chemistry	11	11/11/2004	90	A
1004	Gita	24/09/1988	B3		8	11/11/2004	78	B
1002	Shyam	23/06/1988	P3	Basic Physics	13	11/11/2004	67	C
1005	Sunita	14/09/1987	P3	Basic Physics	13	11/11/2004	78	B
1003	Sita	23/10/1987	B4		5	11/11/2004	67	C
1005	Sunita	13/03/1990	H6		4	11/11/2004	56	D
1004	Gita	21/08/1987	M4	Basic Maths	7	11/11/2004	78	B

Anomaly

- **Insert Anomaly**: We cannot insert prospective course which does not have any registered student or we cannot insert student details that is yet to register for any course.
- **Update Anomaly**: if we want to update the course M4's name we need to do this operation three times. Similarly we may have to update student 1003's name twice if it changes.
- **Delete Anomaly**: if we want to delete a course M4 , in addition to M4 occurs details , other critical details of student also will be deleted. This kind of deletion is harmful to business. Moreover, M4 appears thrice in above table and needs to be deleted thrice.
- **Duplicate Data:** Course M4's data is stored thrice and student 1002's data stored twice .This redundancy will increase as the number of course offerings increases.

Process of normalization

- **Determinant**
  - attribute X decides attribute Y
  - Example: In RESULT relation, Marks attribute may decide the grade attribute .
    - This is represented as Marks->grade and read as Marks decides Grade. But Marks attribute is not a key attribute

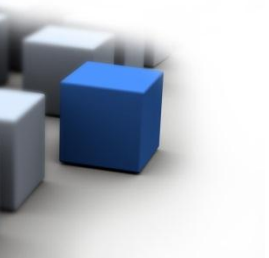


Functional Dependency

- Attribute Y is functional dependent on attribute X if each value of X determines exactly one value of Y
  - Course# -> CourseName
  - Course#-> IName(Assuming one course is taught by one and only one instructor)
  - IName -> Room# (Assuming each instructor has his /her own and non shared room)
  - Marks ->Grade
  - Student#,Course# together (called composite attribute) defines EXACTLY ONE value of marks .This can be symbolically represented as Student#Course# Marks
- This type of dependency is called **functional dependency**.

Full functional dependency

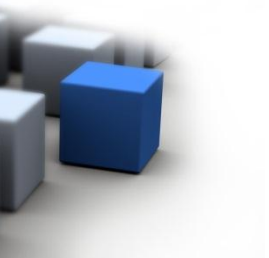
- Marks is fully functional dependent on *student#Course#* and not on the sub set of *Student#Course#*
- CourseName is not fully functionally dependent on student#course#



Partial Dependency

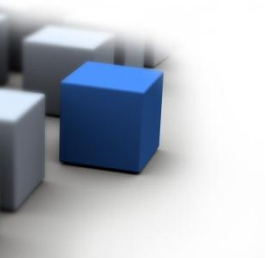
- In the above relationship
  - CourseName
  - Iname
  - Room#
    - are partially dependent on composite attribute
  - Student#Course#
    - because Course# alone can defines the coursename, IName,Room#.

- In a given relation R, Attribute Y is partially dependent on the attribute X only if it is dependent on subset attribute X .



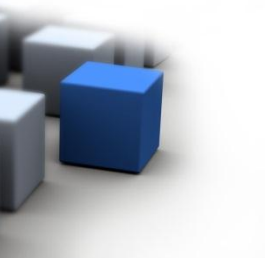
Transitive Dependency

- Room# depends on IName and in turn depends on Course#
- Grade depends on Marks,in turn Marks depends on Student#Course# hence Grade **Fully transitively** depends on Student#Course#



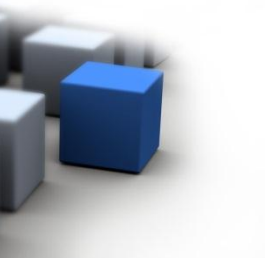
Types of Normal Forms

- **First Normal Form(1NF)**
  - A relation R is said to be in first normal form (1NF) if and only if all the attributes of the relation R, are atomic in nature.



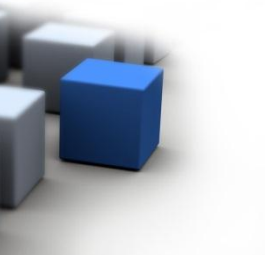
Second Normal Form (2NF)

- It is in the first normal form ,and
- No partial dependency exists between non-key attributes and key attributes.



- In 1NF table structure.
  - Student# is key attribute for Student ,
  - Course# is key attribute for Course
  - Student#Course# together form the composite key attributes for result relationship.

- To make this table 2NF compliant, we have to remove all the partial dependencies.
  - StudentName and DateOfBirth depends only on student#.
  - CourseName , PreRequisite and DurationInDays depends only on Course#
  - DateOfExam depends only on Course#



Tables in 2NF

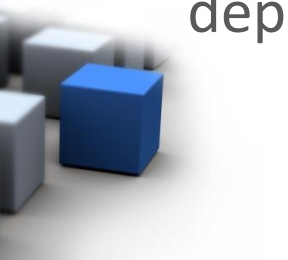
Student #	Student Name	DateofBirth
1001	Ram	Some value
1002	Shyam	Some value
1003	Sita	Some value
1004	Geeta	Some value
1005	Sunita	Some value

Course#	CourseName	Duration of days
C3	Bio Chemistry	3
B3	Botany	8
P3	Nuclear Physics	1
M4	Applied Mathematics	4
H6	American History	5
B4	Zoology	9

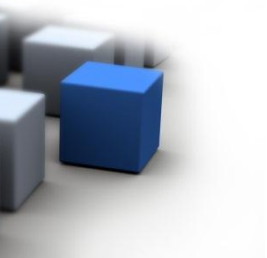
Student#	Course#	Marks	Grade
1001	M4	89	A
1002	M4	78	B
1001	H6	87	A
1003	C3	90	A
1004	B3	78	B
1002	P3	67	C
1005	P3	78	B
1003	B4	67	C
1005	H6	56	D
1004	M4	78	B

Course#	DateOfExam
M4	Some value
H6	Some value
C3	Some value
B3	Some value
P3	Some value
B4	Some value

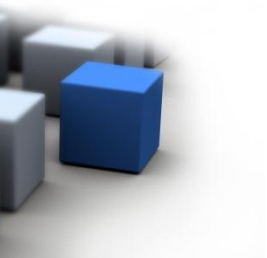
- In the first table (STUDENT) ,the key attribute is Student# and all other non-key attributes, StudentName and DateOfBirth are fully functionally dependant on the key attribute.
- In the Second Table (COURSE) , Course# is the key attribute and all the non-key attributes, CourseName, DurationInDays are fully functional dependant on the key attribute.
- In third table (RESULT) Student#Course# together are key attributes and all other non key attributes, Marks and Grade are fully functional dependant on the key attributes.
- In the fourth Table (EXAM DATE) Course# is the key attribute and the non key attribute ,DateOfExam is fully functionally dependant on the key attribute.



- As per the university evaluation policy,
  - Students who score more than or equal to 80 marks are awarded with “A” grade
  - Students who score more than or equal to 70 marks up till 79 are awarded with “B” grade
  - Students who score more than or equal to 60 marks up till 69 are awarded with “C” grade
  - Students who score more than or equal to 50 marks up till 59 are awarded with “D” grade

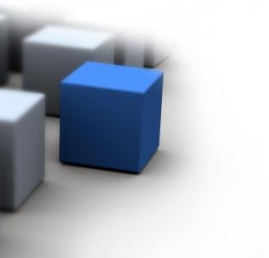


- In the present RESULT table structure ,
  - We don’t have an option to introduce new grades like A+ ,B- and E
  - We need to do multiple updates on the existing record to bring them to new grading definition
  - We will not be able to take away “D” grade if we want to.
  - 2NF does not take care of all the anomalies and inconsistencies.



Third Normal Form (3NF)

- It is in 2NF
- No transitive dependency exists between non-key attributes and key attributes.
  - The grade attribute is dependant on “Marks “ and in turn “Marks” is dependent on Student#Course#.



Student#	Course#	Marks
1001	M4	89
1002	M4	78
1001	H6	87
1003	C3	90
1004	B3	78
1002	P3	67
1005	P3	78
1003	B4	67
1005	H6	56
1004	M4	78

UpperBound	LowerBound	Grade
100	95	A+
94	90	A
89	85	B+
84	80	B
79	75	B-
74	70	C
69	65	C-