

DenId	DenName	PatNo	PatName	AppNo	AppDate	NurNo	NurName	Procedure	Cost
D1	Smith	P1	Jones	A1	1/02/09	N1	Green	Filling	300
D1	Smith	P1	Jones	A2	1/03/09	N1	Green	Filling	300
D1	Smith	P2	Patel	A3	3/01/09	N2	Sarah	Canal	1000
D2	Barky	P3	Thom	A4	6/03/09	N3	Brown	Crown	700
D3	Thomas	P4	Ronnie	A5	8/06/09	N4	Cann	Filling	300
D4	Sam	P5	Emily	A6	9/07/09	N1	Green	Canal	1000
D2	Barky	P6	Sana	A7	8/07/09	N2	Sarah	Canal	1000

```

new 1 x
1 D1, Smith, P1, Jones, A1, 1/02/09, N1, Green, Filling, 300
2 D1, Smith, P1, Jones, A2, 1/03/09, N1, Green, Filling, 300
3 D1, Smith, P2, Patel, A3, 3/01/09, N2, Sarah, Canal, 1000
4

```

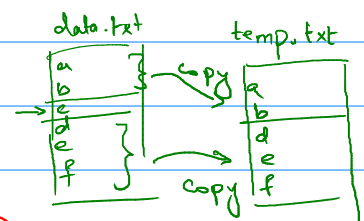
```

line = fin.getline( );
words = tokenize (line, ',')           returns an array of strings
record = new Appointment();
record.docID = words[0];                ["This", "is", "a", "book"]
record.docName = words[1];
record.patID = words[2];
;
record.cost = toInteger(words[9]);
allAppointments.add(record);

```

Write a C++ Program that

1. Reads the file & displays the appointments
2. Display with record
3. Find & Display record by doctor name
4. Sort the data according to given column
5. Show names of doctors who earn more than 1000
6. Show how many appointments for less than 500
7. Add a new record & save to file
8. Delete a record & save to file
9. Change the value of cost in whole file



1. create temp file
2. copy data before sort
3. copy data after sort
4. delete data.txt
5. rename temp.txt to data.txt

## Delimited Records

Use a marker or delimiter to separate the data of different fields of the records  
, space tab ; \$

```
new 1 x
1 D1, Smith, P1, Jones, A1, 1/02/09, N1, Green, Filling, 300
2 D1, Smith, P1, Jones, A2, 1/03/09, N1, Green, Filling, 300
3 D1, Smith, P2, Patel, A3, 3/01/09, N2, Sarah, Canal, 1000
4
```

- What happens when delimiter is part of data
- How to skip records

## Fixed Length Records : Size of every field is fixed

```
new 1 x new 2 x
1 D1 Smith P1 Jones A1 1/02/09 N1 Green Filling 300
2 D1 Smith P1 Jones A2 1/03/09 N1 Green Filling 300
3 D1 Smith P1 Patel A3 3/02/09 N2 Sarah Canal 300
4
```

C++ → seekg()  
tellp() / seekp()

## Problems with File Based Storage Systems

DenId	DenName	PatNo	PatName	AppNo	AppDate	NurNo	NurName	Procedure	Cost	DocNo
D1	Smith	P1	Jones	A1	1/02/09	N1	Green	Filling	300	
D1	Smith	P1	Jones	A2	1/03/09	N1	Green	Filling	300	
D1	Smith	P2	Patel	A3	3/01/09	N2	Sarah	Canal	1000	
D2	Barky	P3	Thom	A4	6/03/09	N3	Brown	Crown	700	
D3	Thomas	P4	Ronnie	A5	8/06/09	N4	Cann	Filling	300	
D4	Sam	P5	Emily	A6	9/07/09	N1	Green	Canal	1000	
D2	Barky	P6	Sana	A7	8/07/09	N2	Sarah	Canal	1000	

↑  
DentPhone
↑  
PatPhone

1. Code is dependent on shape / structure of data
2. Different users need access <sup>only</sup> to different sections of data
3. Sharing of Data
4. Data Security
5. Data Duplication
6. Data Inconsistency
7. Backup & Recovery Procedures -
8. Handling connection / relationship b/w data
9. Handling Adhoc Queries
10. Enforcing Standards

Database  $\rightarrow$  Relational Schema

Relation  $\swarrow$  table

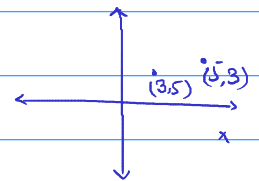
$\hookrightarrow$  Set of tuples

Set: Collection of objects

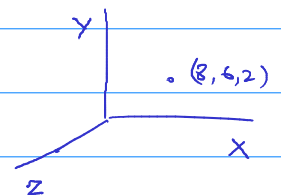
	Ordered Data	Duplication
Array	✓	✓
Bag	✗	✓
Set	✗	✗

Ordered Pair

$P_1 = (3, 5)$   
 $\swarrow$  value along x axis       $\searrow$  value along y axis



$P_2 = (5, 3)$        $P_1 \neq P_2$



$P_3 = (8, 6, 2)$       tuple

$\langle x, y, z \rangle$   $\swarrow$  tuple

An ordered list containing  $k$  values such that

1st item represents value for 1st variable

2nd "      2nd "

3rd "      3rd "

$\vdots$

kth "      kth "

$a = \langle a_1, a_2, a_3, \dots, a_k \rangle$

$b = \langle b_1, b_2, b_3, \dots, b_k \rangle$

ith value of the tuple can be any valid value for the ith variable

	A	B	C
$A = \{a, b, c\}$	$T_1 = \langle a, 3, \alpha \rangle$		✓
$B = \{1, 2, 3\}$	$T_2 = \langle c, 2, \beta \rangle$		✓
$C = \{\alpha, \beta, \theta\}$	$T_3 = \langle a, \beta, \Delta \rangle$		X

Name: {Ali, Fatima, Ayyaz, Ayesha, Saleem}  
 DOB: {1/10/20, 2/5/20, 2/10/20, 3/11/19}  
 Gender: {M, F}

$T_1 = \langle \text{Ali}, 1/10/20, M \rangle$   
 $T_2 = \langle \text{Fatima}, 2/5/20, F \rangle$   
 $T_3 = \langle \text{Ali}, 2/10/20, M \rangle$   
 $T_4 = \langle \text{Ayyaz}, 1/10/20, M \rangle$   
 $T_5 = \langle \text{Ayesha}, 3/11/19, F \rangle$

People Name	DOB	Gender
Ali	1/10/20	M
Fatima	2/5/20	F
Ali	2/10/20	M
Ayyaz	1/10/20	M
Ayesha	3/11/19	F

Relation → People = {  $\langle \text{Ali}, 1/10/20, M \rangle$ ,  $\langle \text{Fatima}, 2/5/20, F \rangle$ ,  $\langle \text{Ali}, 2/10/20, M \rangle$ ,  $\langle \text{Ayyaz}, 1/10/20, M \rangle$ ,  $\langle \text{Ayesha}, 3/11/19, F \rangle$  }

### Relation State

The state of the data at a given time  
 It can change over time

## Constraints

1. Implicit Constraints → whole record/tuple cannot be duplicate or repeated
2. Explicit Constraints (Syntax Rules)
3. Semantic Constraints (Business Rules)

### Explicit Constraints

Domain Constraint: The set of <sup>valid</sup> values that can appear as attribute value

Required / optional :

Unique

Default Value

Key

Integrity constraint { Superkey: Any set of attributes that can be used to recognize every record  
Candidate key: Superkeys which contain the least # of attributes  
Primary key: One of the candidate keys  
Default Superkey: Set of all attributes of the relation (by Implicit Constraint)

Referential Integrity constraint

Foreign key: pointing to a valid value in a remote file/relation

CURD	domain	required	unique	default	integrity constraint	referential integrity
Create	✓	✓	✓	✓	✓	✓
Update	✓	✓	✓	✓	✓	✓ (2 cases)
Read	✗	✗	✗	✗	✗	✗
Delete	✗	✗	✗	✗	✗	✓

1. Name
2. Roll #
3. Course & Section
4. Date
5. What is special about today's date / quiz.