www.nammakalvi.in

J. BASKARAN M.Sc., B.Ed. (C.S)
jbaskaran89@gmail.com
Puducherry.

J. ILAKKIA M.Sc., M.Phil., B.Ed. (C.S)
jilakkia@gmail.com
Puducherry.

2. DATA ABSTRACTION

Section – A

Choose the best answer **(1 Mark)**

1. Which of the following functions that build the abstract data type ?

(A) Constructors (B) Destructors (C) recursive (D)Nested

2. Which of the following functions that retrieve information from the data type?

(A) Constructors **(B) Selectors** (C) recursive (D)Nested

3. The data structure which is a mutable ordered sequence of elements is called

(A) Built in **(B) List** (C) Tuple (D) Derived data

4. A sequence of immutable objects is called

(A) Built in (B) List **(C) Tuple** (D) Derived data

5. The data type whose representation is known are called

(A) Built in datatype (B) Derived datatype

(C) Concrete datatype (D) Abstract datatype

6. The data type whose representation is unknown are called

(A) Built in datatype (B) Derived datatype

(C) Concrete datatype **(D) Abstract datatype**

7. Which of the following is a compound structure?

(A) Pair (B) Triplet (C) single (D) quadrat

8. Bundling two values together into one can be considered as

(A) Pair (B) Triplet (C) single (D) quadrat

9. Which of the following allow to name the various parts of a multi-item object?

(A) Tuples (B) Lists **(C) Classes** (D) quadrats

10. Which of the following is constructed by placing expressions within square brackets?

(A) Tuples **(B) Lists** (C) Classes (D) quadrats

Section-B

Answer the following questions **(2 Marks)**

1. What is abstract data type?

- **Abstract Data type (ADT)** is a type or class for objects whose behavior is defined by a set of value and a set of operations.

2. Differentiate constructors and selectors.

CONSTRUCTORS	SELECTORS
• Constructors are functions that build the abstract data type.	• Selectors are functions that retrieve information from the data type.
• Constructors create an object, bundling together different pieces of information	• Selectors extract individual pieces of information from the object.

3. What is a Pair? Give an example.

- Any way of bundling two values together into one can be considered as a pair.
- Lists are a common method to do so.
- Therefore List can be called as Pairs.
- **Example:** lst[(0,10),(1,20)]

4. What is a List? Give an example.

- List can store multiple values of any type.
- List is constructed by placing expressions within square brackets separated by commas.
- Such an expression is called a list literal.
- **Example:** lst[10,20]

5. What is a Tuple? Give an example.

- A tuple is a comma-separated sequence of values surrounded with parentheses.
- Tuple is similar to a list.
- Cannot change the elements of a tuple.
- **Example:** Color= ('red', 'blue', 'Green')

Section-C

Answer the following questions **(3 Marks)**

1. Differentiate Concrete data type and abstract datatype.

CONCRETE DATA TYPE	ABSTRACT DATA TYPE
• Concrete data types or structures (CDT's) are direct implementations of a relatively simple concept.	• Abstract Data Types (ADT's) offer a high level view (and use) of a concept independent of its implementation.
• A concrete data type is a data type whose representation is known.	• Abstract data type the representation of a data type is unknown.

2. Which strategy is used for program designing? Define that Strategy.

- A powerful strategy for designing programs is **'wishful thinking'**.
- Wishful Thinking is the formation of beliefs and making decisions according to what might be pleasing to imagine instead of by appealing to reality.

3. Identify Which of the following are constructors and selectors?

(a) N1=number() -- **Constructor**

(b) accetnum(n1) -- **Selector**

(c) displaynum(n1) -- **Selector**

(d) eval(a/b) -- **Selector**

(e) x,y= makeslope (m), makeslope(n) -- **Constructor**

(f) display() -- **Selector**

4. What are the different ways to access the elements of a list. Give example.

- The elements of a list can be accessed in two ways.

1. Multiple Assignment:

- Which unpacks a list into its elements and binds each element to a different name.

Example:

```
lst := [10, 20]
x, y := lst
```

- ***x*** will become10 and ***y*** will become 20.

2. Element Selection Operator:

- It is expressed using square brackets.
- Unlike a list literal, a square-brackets expression directly following another expression does not evaluate to a list value, but instead selects an element from the value of the preceding expression.

Example:

```
lst[0]
10
lst[1]
20
```

5. Identify Which of the following are List, Tuple and class ?

(a) arr [1, 2, 34]	--	**List**
(b) arr (1, 2, 34)	--	**Tuple**
(c) student [rno, name, mark]	--	**Class**
(d) day= (‘sun’, ‘mon’, ‘tue’, ‘wed’)	--	**Tuple**
(e) x= [2, 5, 6.5, [5, 6], 8.2]	--	**List**
(f) employee [eno, ename, esal, eaddress]	--	**Class**

Section - D

Answer the following questions: **(5 Marks)**

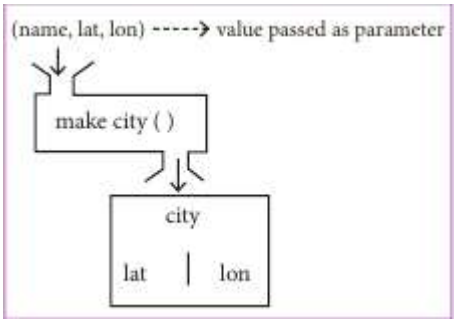
1. How will you facilitate data abstraction. Explain it with suitable example.

- Data abstraction is supported by defining an abstract data type (ADT), which is a collection of constructors and selectors.
- To facilitate data abstraction, you will need to create two types of functions:
  - **Constructors**
  - **Selectors**

a) Constructor:

- Constructors are functions that build the abstract data type.
- Constructors create an object, bundling together different pieces of information.
- For example, say you have an abstract data type called city.

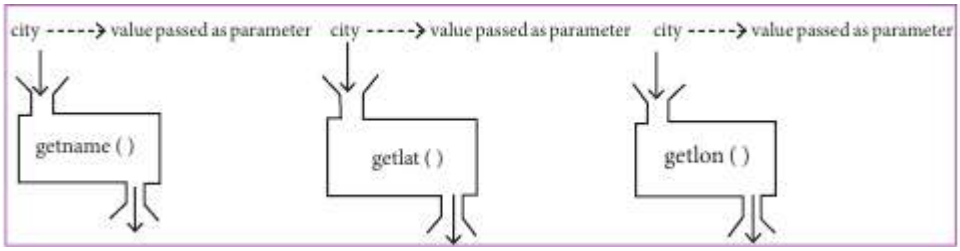
- This city object will hold the city's name, and its latitude and longitude.
- To create a city object, you'd use a function like **city = makecity (name, lat, lon).**
- Here makecity (name, lat, lon) is the constructor which creates the object city.



b) Selectors:

- Selectors are functions that retrieve information from the data type.
- Selectors extract individual pieces of information from the object.
- To extract the information of a city object, you would use functions like
  - **getname(city)**
  - **getlat(city)**
  - **getlon(city)**

These are the selectors because these functions extract the information of the city object.



2. What is a List? Why List can be called as Pairs. Explain with suitable example.

LIST:

- List is constructed by placing expressions within square brackets separated by commas.
- Such an expression is called a list literal.
- List can store multiple values.
- Each value can be of any type and can even be another list.
- The elements of a list can be accessed in two ways.

1. Multiple Assignment:

- Which unpacks a list into its elements and binds each element to a different name.

Example:

```
lst := [10, 20]
x, y := lst
```

- ***x*** will become10 and ***y*** will become 20.

2. Element Selection Operator:

- It is expressed using square brackets.
- Unlike a list literal, a square-brackets expression directly following another expression does not evaluate to a list value, but instead selects an element from the value of the preceding expression.

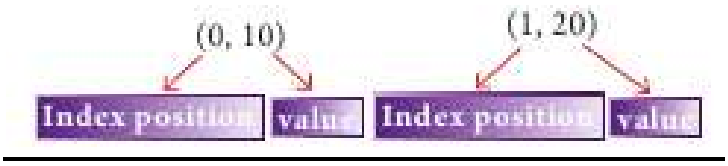
Example:

```
lst[0]
10
lst[1]
20
```

PAIR:

- Any way of bundling two values together into one can be considered as a pair.
- Lists are a common method to do so.
- Therefore List can be called as Pairs.

Example: lst[(0,10),(1,20)]



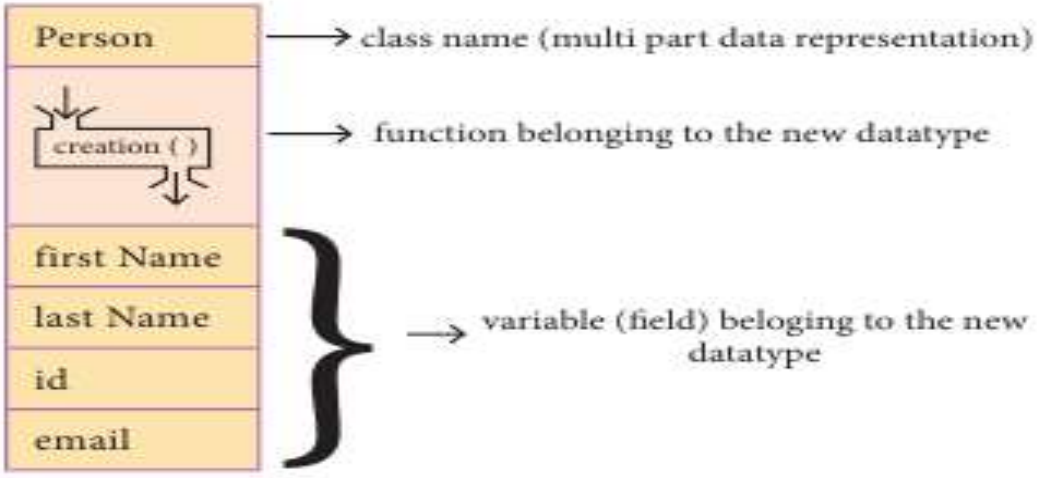
3. How will you access the multi-item. Explain with example.

MULTI-ITEM:

- The structure construct in OOP languages it's called **class construct** is used to represent multi-part objects where each part is named.
- Consider the following pseudo code:

```
class Person:
    creation( )
    firstName := " "
    lastName := " "
    id := " "
    email := " "
```

- The new data type Person is pictorially represented as,



Let main() contains	
p1:=Person()	statement creates the object.
firstName := " Padmashri "	setting a field called firstName with value Padmashri
lastName :="Baskar"	setting a field called lastName with value Baskar
id :="994-222-1234"	setting a field called id value 994-222-1234
email="compsci@gmail.com"	setting a field called email with value compsci@gmail.com
- - output of firstName : Padmashri	

- The class (structure) construct defines the form for multi-part objects that represent a person.
- Person is referred to as a class or a type, while p1 is referred to as an object or an instance.
- Using class you can create many objects of that type.
- Class defines a data abstraction by grouping related data items.
- A class as bundled data and the functions that work on that data that is using class we can access multi-part items.

PREPARED BY

J. BASKARAN M.Sc., B.Ed. (C.S)
jbaskaran89@gmail.com
Puducherry.

J. ILAKKIA M.Sc., M.Phil., B.Ed. (C.S)
jilakkia@gmail.com
Puducherry.