key constraints

one to many, many to many etc. the direction of the arrow is pointed where 'one' refers, attribute types :- composite attributes.multivalued attributes.derived attributes. Generalization : composing two or more entities together. specialization reverse of generalization.
Disjoint :- user can be a member of at most one entity. overlap just opposite. total atleast one. disjoint can be present in both.partial.

Participation Constraints

total or partial. represented by bold lines.

superkey and candidate key

superkey one or more attributes together. candidate key is a minimal superkey. an entity that has a primary key is called as a strong entity. the entity whose primary key is being used is called as the identifying owner.

JDBC Application Programming

Client - Server Architecture . 2 tier and 3 tier architecture.
ODBC: Open Database Connectivity.JDBC located in java.sql package.

JDBC Application Programming (cont)

JDBC-ODBC bridge • Con: ODBC must be installed
• JDBC database client • Con: JDBC driver for each server must be available
• JDBC middleware client • Pro: Only one JDBC driver is required • Application does not need direct connection

JDBC Steps

1. Load the driver 2. Define the Connection URL 3. Establish the Connection 4. Create a Statement object 5. Execute a query 6. Process the results 7. Close the Connection
commit () rollback() getMetaDataObject getwarnings().
ResultSetMetaData answers the following questions:
• How many columns are in the result set?
• What is the name of a given column?
• Are the column names case sensitive?
• What is the data type of a specific column?
• What is the maximum character size of a column?

JDBC Steps (cont)

• Can you search on a given column?

Query

insert into table name values (select statement). Aggregation [MAX,MIN,AVG,COUNT,SUM]
SELECT product, Sum(price*quantity) AS TotalSales
FROM Purchase
WHERE date > “10/1”
GROUP BY product
without group by
SELECT DISTINCT x.product,
(SELECT Sum(y.price*y.quantity)
FROM Purchase y
WHERE x.product = y.product
AND y.date > ‘10/1’)
AS TotalSales
FROM Purchase x
WHERE x.date > “10/1”

Joins

left outer join:For tables A and B, contains all records of the "left" table
(A), even if the join-condition does not find any matching record in the "right" table

OORDBMS

Abstraction: ignoring the parts that are not important.focus on what an object is and what it does rather than how it is done.
Encapsulation: information hiding.separating external aspects from the internal implementation.
Class: A group of objects with the same attributes and methods.
Methods
1.**Member Method:** defined on Instance Data
2.**Static Method** : invoked on the object type.can be used that are global.
3.**Constructor Method**: Built in constructor method.

Methods to compare objects: Member Method

Define a special kind of member methods to compare objects.
☐ Define either a map method or an order method in an object type.
☐ Map Method
• Map object instances into one of the scalar types DATE, CHAR, NUMBER,…

Class Concepts

Subclass: A class of objects that is defined as a special case of a more general class,the process of forming subclasses is called specialization.
Superclass: A class of objects that is defined as a general case of a number of special classes (the process of forming a superclass is called generalization). All instances of a subclass are also instances of its superclass.
Inheritance: By default, a subclass inherits all the properties of its superclass (or it can redefine some (or all) of the inherited methods). Additionally, it may define its own unique properties.

Oracle Methods

Member Methods
CREATE OR REPLACE TYPE BODY person_type AS
MEMBER FUNCTION get_areacode RETURN VARCHAR2 IS
BEGIN
RETURN SUBSTR(phone, 1, 3);

Oracle Methods (cont)

END get_areacode;
END;
/SELECT c.contact.get_areacode()
FROM contacts c;
-- Constructor Method
Every object type has a constructor method implicitly defined by
system.
☐ Returns a new instance of the user-defined object type and sets up the
values of its attributes.
☐ The name of constructor method is the same as the name of the object
type.
p = person_type('Scott Tiger', '321-123-1234');

Creating Object Table

CREATE TABLE person_table OF person_type;
INSERT INTO person_table VALUES (person_type ('Scott Tiger', '321-123-1234'));
SELECT VALUE(p) FROM person_table p WHERE p.name = 'Scott

overloading/overriding

CREATE TYPE Shape_typ AS OBJECT (...,
MEMBER PROCEDURE Enlarge(x
...) NOT FINAL; /
CREATE TYPE Circle_typ UNDER
MEMBER PROCEDURE Enlarge(x
NUMBER),
Shape_typ (...,
CHAR(1))); /
--Define the inherited method Enlarge() to deal
--input parameters.
with different types of
CREATE TYPE Shape_typ AS OBJECT (...,
MEMBER PROCEDURE Area(),
FINAL MEMBER FUNCTION id(x NUMBER)...
) NOT FINAL; /
CREATE TYPE Circle_typ UNDER Shape_typ (...,
OVERRIDING MEMBER PROCEDURE Area(),
...); /

Order Methods

CREATE TYPE
circle_type x
y
r
AS OBJECT (
NUMBER,
NUMBER,
NUMBER,
ORDER MEMBER FUNCTION match(c circle_type)

Order Methods (cont)

CREATE OR REPLACE TYPE BODY circle_type AS
ORDER MEMBER FUNCTION match (c circle_type)
BEGIN
RETURN INTEGER); /
RETURN INTEGER IS
IF r < c.r THEN
RETURN –1;
-- 3.14*r2 < 3.14*c.r2
-- any negative number
THEN
-- any positive number
ELSIF r > c.r
RETURN 1;
ELSE
RETURN 0;
END IF;
END;
END;
SELECT FORM FROM circles c
WHERE VALUE(c) < (circle_type(40, 25, 5)) ;