Lecture Notes – Pattern Matching

Section 1: Lecture Summary

Pattern matching is a filtering technique used in the WHERE clause to search for records that match specific string patterns. It uses the LIKE operator combined with wildcard characters to define flexible search criteria. This method is particularly useful for finding employees by job title characteristics, filtering customers by email domains, or identifying records based on string length and composition.

Section 2: Key Concepts and Explanations

Wildcard Characters

The percent sign (%) represents zero or more characters of any type. It allows flexible matching at any position within a string. For example, "%Manager" matches any string ending with Manager, while "A%" matches strings beginning with the letter A.

The underscore (_) represents exactly one character of any type. Multiple underscores can be combined to specify exact string lengths. For instance, "_____" (five underscores) matches any string with exactly five characters.

Pattern Matching Syntax

The basic structure uses the LIKE operator: `ColumnName LIKE 'pattern'`

Patterns can combine wildcards for precise matching. Common applications include checking email domains with "%@gmail.com%" to ensure both an @ symbol and .com extension, or using "%.___" to match three-letter file extensions.

Mixing Wildcards

Wildcards can be combined in a single pattern for more sophisticated searches. For example, "%@gmail.com" ensures an email contains @ followed by gmail.com at the end, and "%Q%" finds any string containing the letter Q anywhere within it.

Section 3: Example Code and Use Cases

Example 1: Find Employees with Manager in Job Title

```
SELECT EmpID, FirstName, JobTitle
FROM Employees
WHERE JobTitle LIKE '%Manager%';
```

This query returns all employees whose JobTitle contains the word Manager, such as HR Manager, Finance Manager, or Marketing Manager. The percent signs allow for any characters before and after Manager.

Example 2: Find Customers with Gmail Email Addresses

```
SELECT CustomerID, FirstName, Email
FROM Customers
WHERE Email LIKE '%@gmail.com%';
```

This retrieves all customers whose email addresses contain the Gmail domain. The pattern ensures the @ symbol exists followed by gmail.com.

Example 3: Find Customers with Exactly Five-Letter Last Names

```
SELECT CustomerID, FirstName, LastName, Email
FROM Customers
WHERE LastName LIKE '_____';
```

This query selects customers whose LastName has exactly five characters. Each underscore represents one character position, so five underscores match any five-letter name.

Example 4: Find Products Starting with Specific Letter

```
SELECT ProductID, ProductName, Price
FROM Products
WHERE ProductName LIKE 'S%';
```

This returns all products whose ProductName begins with the letter S, followed by any number of characters.

Example 5: Find Orders from Customers in Specific Cities

```
SELECT OrderID, CustomerID, OrderDate, TotalAmount
FROM Orders
WHERE CustomerID IN (
  SELECT CustomerID FROM Customers WHERE City LIKE '%York%'
);
```

This uses pattern matching to find customers whose City contains "York" and returns their orders.

Section 4: Key Takeaways

Pattern matching with LIKE and wildcards provides flexible string filtering without requiring exact matches. The percent sign (%) matches variable-length character sequences, while the underscore (_) matches individual characters with predictable

positions. These tools are essential for real-world database queries where exact string values may be unknown or where partial matching is desired. Combining multiple wildcards in a single pattern enables sophisticated filtering for email validation, name searches, and other string-based conditions.