

3-schichtige Informationssystem-Architektur

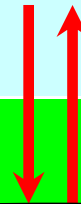
- *plattformunabhängig*
- *beliebige Endgeräte*
- *Client als Applikation bzw. als Applet*



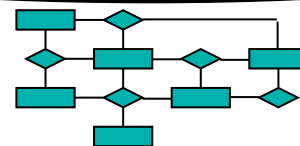
- *plattformunabhängig*
- *objektorientierte Architektur*

Server: Anwendungslogik

hier einfach Dateisystem des PCs
jetzt: DB-Anschluss



strukturierte Daten



Java-Abbild von DB-Inhalten

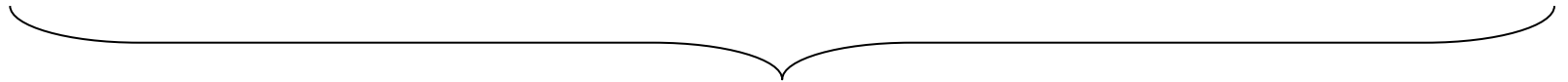
Java

```

<<class>>
User
name: String
email: String
password: String
...
    
```

```

User u = new User();
u.setName („Heinz Fassbender“)
u.setEmail („fassbender@fh-aachen.de“)
u.setPassword („geheim“)
    
```

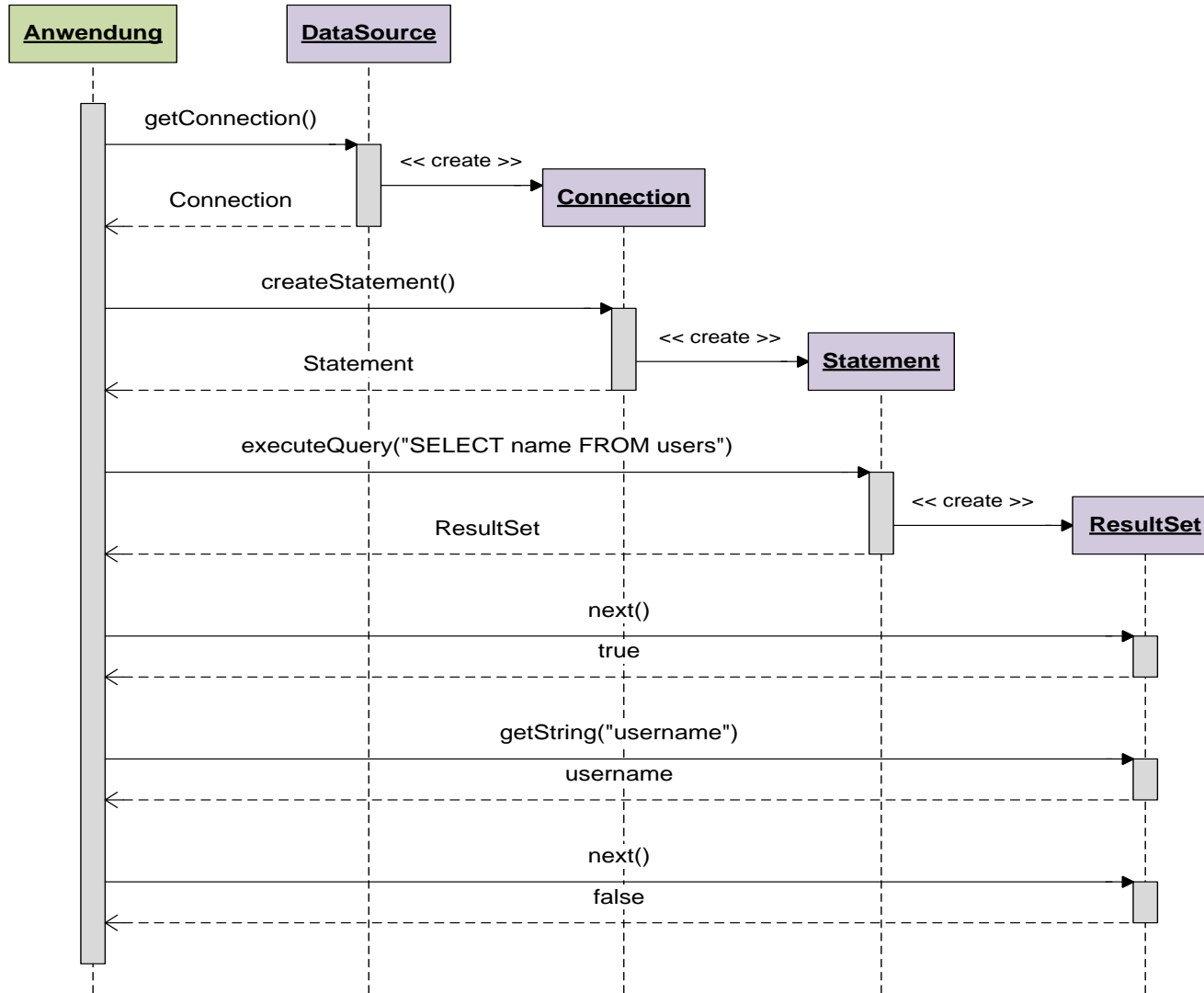


DB

Users		
name	email	password
Heinz Fassbender	fassbender@fh-aachen.de	geheim
...

Pro Zeile: 1 Objekt
 Ganze Tabelle: ResultSet der Objekte

Ablauf einer SQL-Anfrage



Absetzen von SQL-Statements allg.

```
DataSource dataSource = ... //siehe Folie 6
Connection connection;
try {

    connection = dataSource.getConnection();

    Statement statement = connection.createStatement();

    ResultSet resultSet = statement.executeQuery(„SQL“);

    while(resultSet.next()) {

        /* ResultSet verarbeiten */

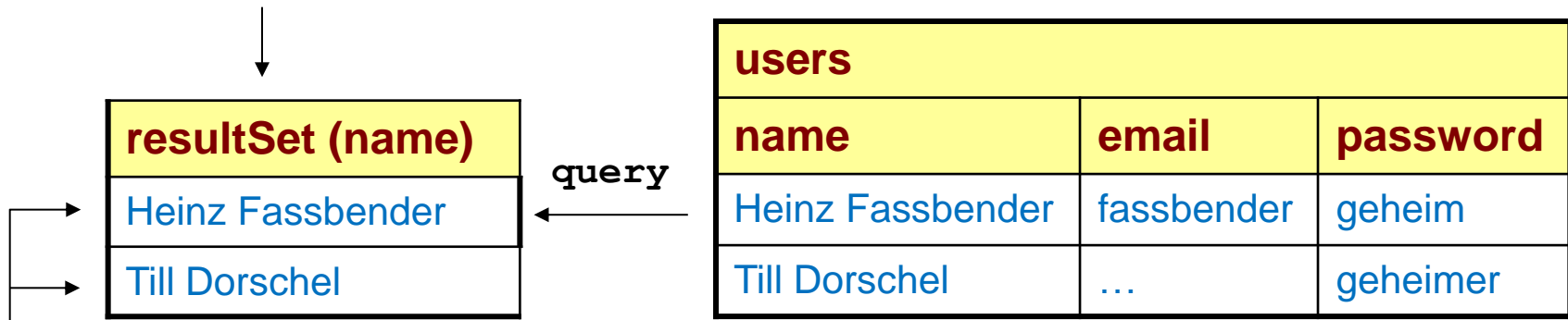
    }
} finally {
    if(connection != null) { connection.close(); }
}
```

Absetzen von SQL-Statements speziell

```

/* Namen aller gespeicherten Benutzer ermitteln */
String query = „SELECT name FROM users“;
ResultSet resultSet = statement.executeQuery(query)

```



```

List<String> names = new ArrayList<String>();
while(resultSet.next()) {
    String name = resultSet.getString(„name“);
    names.add(name);
}

```

```
names: „Heinz Fassbender“; „Till Dorschel“
```

Erstellung einer DataSource

- **Datenquelle zentrale Schnittstelle für Connection-Handling**
- **Konfig wichtig, da sonst Performance Engpass**
- **Verwendung von „pooled“ DataSource**
 - nicht für jede Anfrage neue DB-Verbindung
 - sondern Verwaltung eines Pools von DB-Verbindungen
- **hier folgende primitive Verbindung als DataSource durch "jdbcDriver" für die HSQL-Datenbank**

```
BasicDataSource dataSource = new BasicDataSource();  
dataSource.setDriverClassName(jdbcDriver.class.getName());  
dataSource.setUrl("jdbc:hsqldb:mem:TestSpeicherDB");  
dataSource.setUsername("sa");  
dataSource.setPassword("");
```