



# Databases

- **A persistent storage: a place to store the state of objects**
- **Facilities provided in J2SE and J2EE are not suitable for handheld devices**
- **MIDP provides a record-oriented database mechanism to persistently store data and retrieve it later**



# MIDP'S RMS

- Lightweight record-oriented database
  - Device independent API
  - Unique recordID for each record within the store
  - A record is an array of bytes
  - Shared within MIDlet suite
  - Support for enumeration, sorting, and filtering
- `javax.microedition.rms`



# MIDP - RMS Methods

- **Record Store**  
openRecordStore, closeRecordStore, listRecordStore,  
deleteRecordStore, getRecordSize, getNumRecords
- **Record Data**  
addRecord, deleteRecord, getRecord, setRecord,  
getRecordSize
- **Record Selection**  
RecordEnumeration, RecordFilter, RecordComparator



## RMS: Record Stores

- To open a record store:
  - `RecordStoredb = RecordStore.openRecordStore("myDB", true);`
- To close a record store:
  - `db.closeRecordStore();`



## Create/Add a New Record

- To create a new record:

```
ByteArrayOutputStreambaos= new
```

```
    ByteArrayOutputStream();
```

```
DataOutputStreamdos = new DataOutputStream(baos);
```

```
dos.writeUTF(record);
```

```
Byte b[] = baos.toByteArray();
```

```
db.addRecord(b, 0, b.length);
```