**RecordDatabase**

**Bewertung:**

**-Speicherbedarf:** die RecordDatabase benötigt einen relativ großen Speicherbedarf, weil der String eine feste Länge hat und dadurch benötigen kleine Daten Sätze größere Speichkapazität. Somit steigt der Speicher.

**-Zugriffszeit:** Die Zugriffszeit weist eine geringe Dauer auf, da die Datensätze unmittelbar zugegriffen werden kann, da ihre Länge genau bekannt ist. Somit kann ein Filepointer schnell an sein Ziel gesetzt werden

**-Performanz:** Die Performanz ist langsamer, denn dadurch, dass die Länge bekannt ist, werden dann die Datensätze angezielt.

**-Wartbarkeit:** Es gibt keine offensichtliche Beeinflussung der Daten, weil die Daten in Bytes geschrieben werden

**CSVDatabase**

**Bewertung:**

**-Speicherbedarf:** die CSVDatabase braucht einen niedrigen Speicherbedarf

**-Zugriffszeit:** der Datensatz wird eingelesen, bis dann die Stelle des Datensatzes erreicht ist

**-Performanz:** Sie ist beim deutlich schneller, weil die Datensätze kürzer und in keine Füllbits geschrieben sind. Leider weist es beim Lesen überraschend eine lange Zeit auf, weil es Zeichen für Zeichen abgearbeitet werden muss, damit sie das Trennzeichen lokalisieren kann.

**-Wartbarkeit/Betriebssicherheit:** Es ist sehr einfach die Daten zu beeinflussen, weil die Datei offensichtlich geschrieben wird. Fehler können direkt behoben werden.

**SerializableDatabase**

**Bewertung:**

**-Speicherbedarf:** die SerializableDatabase braucht einen deutlich niedrigen Speicherbedarf

**-Zugriffszeit:** der Datensatz wird eingelesen, bis dann die Stelle des Datensatzes erreicht ist

**-Performanz:** Ist von allen Klassen am schnellsten, weil die Objekte direkt weggeschrieben werden können und mein lesen ebenso schnell, weil das Objekt direkt verfügbar ist

**-Wartbarkeit/Betriebssicherheit:** nicht direkt beeinflussbar, da Sie in Bytes geschrieben werden. Fehler können nur mit Streams behoben werden

**Kann Ihre Software auch das 10^2, 10^5, 10^7 oder 10^9-fache der Daten verarbeiten?**

Nein, weil die Daten direkt auf die Festplatte gelesen wird und dort abgelegt wird. Daher werden wenig Ressourcen vom CPU und RAMs benötigt

**Was müssen Sie implementieren, dass es kein Maximum gibt? Unterscheiden Sie hierbei in die Bereiche Software und Hardware**

Die Festplatte müsste eine noch deutlich höhere Speicherkapazität aufweisen.

Softwaremäßig darf der Verbrauch des Arbeitsspeichers nicht durch Datenmenge verarbeitet werden.