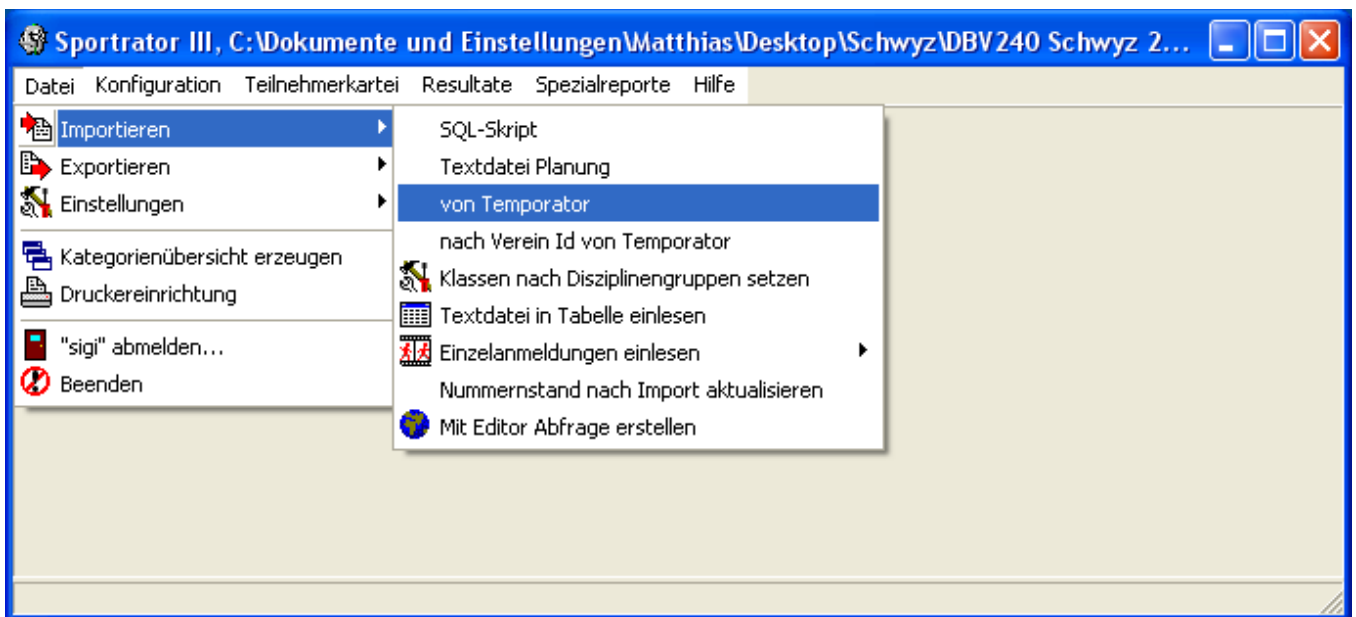


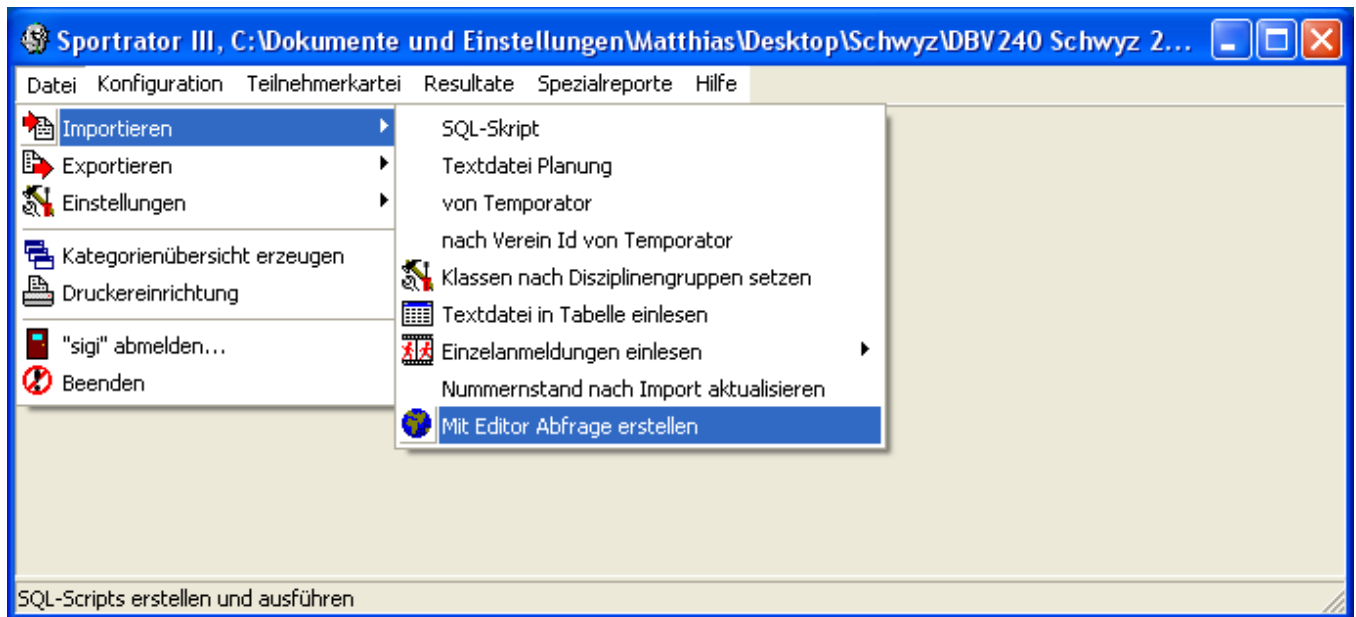
## Erstellen einer Sportrator-Datenbank

### Vereinswettkampf konfigurieren

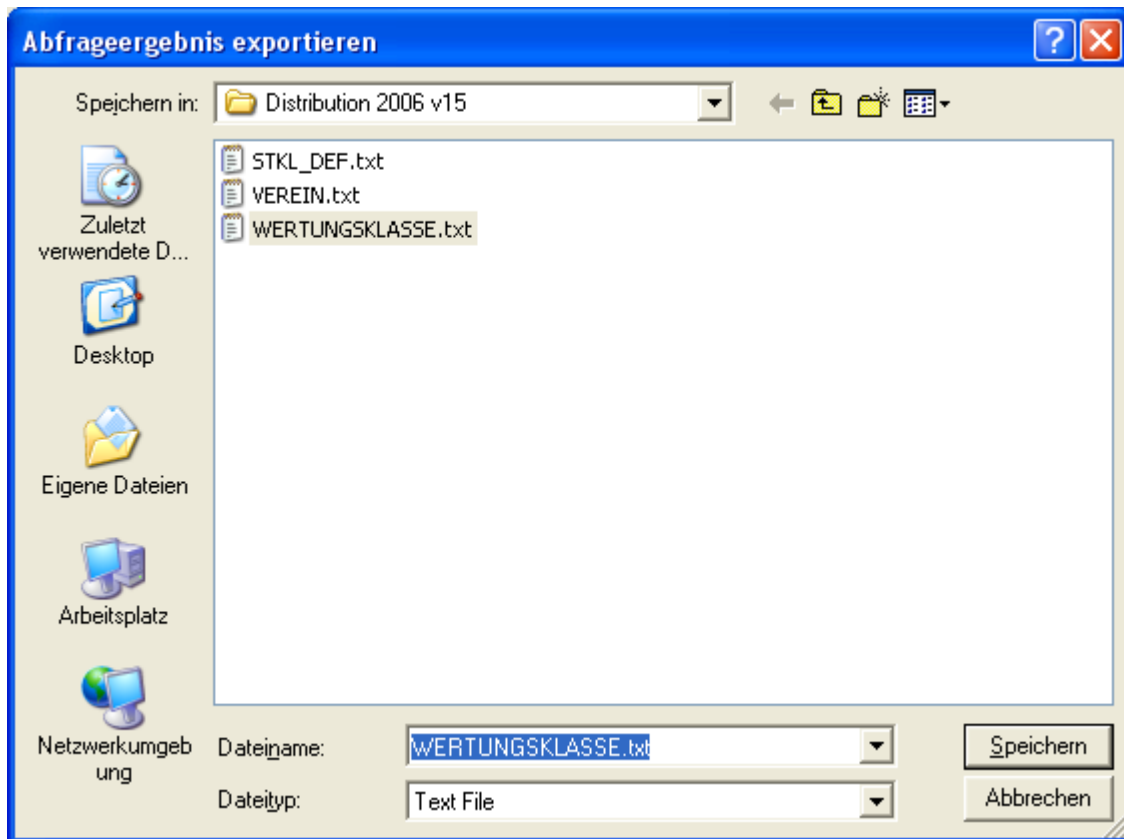
1. Kopie von Leerdatenbank **DBV300LEER mit Benutzern.GDB** erstellen und diese nach dem Anlass benennen, z.B. DBV300\_KTF\_SG2009.gdb
2. Sportrator starten und zu der kopierten Leerdatenbank verbinden.  
Login: admin  
Passwort: data  
(Falls eine alte Datenbank verwendet werden soll, so muss an dieser Stelle das Script **Script 001-V2009 Alle Bewegungsdaten löschen für Temporator Import.sql** ausgeführt werden. Allerdings weicht das Procedere zur Erstellung einer Datenbank mit bestehenden Daten teilweise massiv vom Inhalt dieses Dokumentes ab. Diese Art der Erstellung einer Datenbank ist nur erfahrenen Benutzern zu empfehlen, weil sie nicht allgemein in einem Kochbuch beschrieben werden kann.)
3. Lizenzschlüssel mit der richtigen Bezeichnung für den Anlass erfassen. Danach in der Grundkonfiguration bei jedem dem Administrator untergeordneten Benutzer (pc1- pc10) das Ablaufdatum löschen, (Feld markieren, löschen und speichern) damit sich der Wert entsprechend der Administrator-Lizenz automatisch anpasst.
4. Datei Temporator-Exportdatei **Temp2Spo.csv** in Sportrator einlesen. (Wenn keine Fehlermeldung angezeigt wird, war der Import erfolgreich. Derselbe Import lässt sich nicht mehrmals ausführen.)



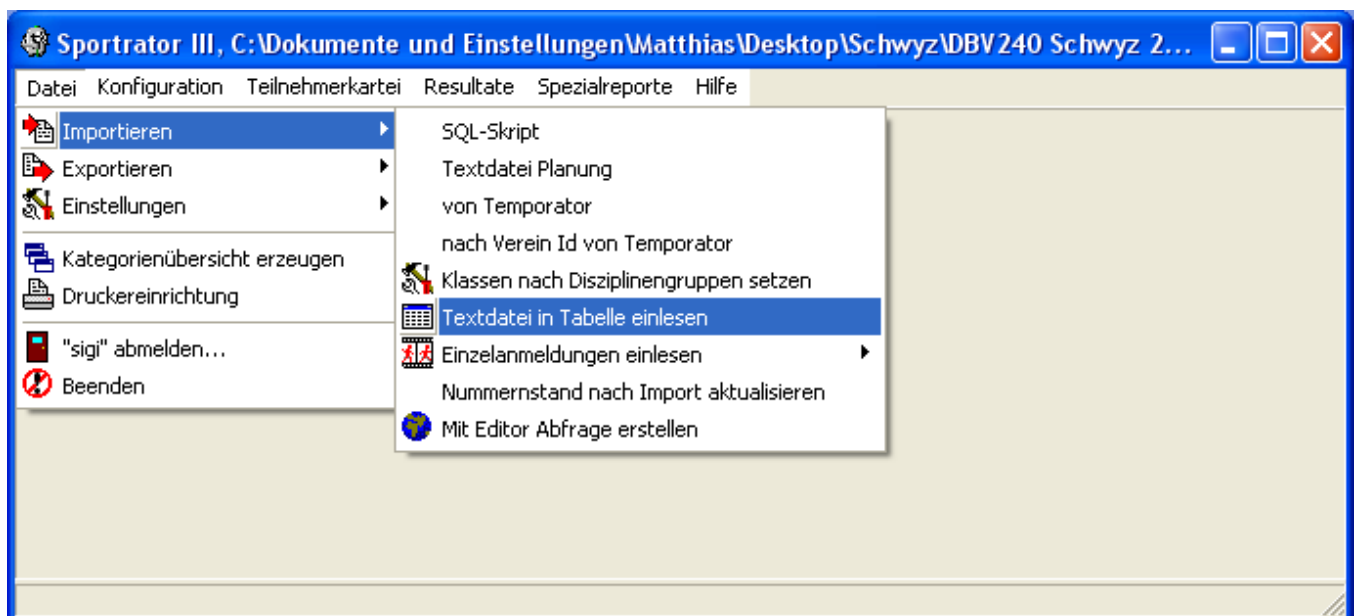
5. Scripteditor öffnen (Datei..Importieren..**Mit Editor Abfrage erstellen**)



6. **Script 005-V2009 Insert Natu STHF-Disz.sql** ausführen.  
Dieses Script kann nur einmalig ausgeführt werden. Es werden vier Zeilen angezeigt, als Quittung für das erfolgreiche Ausführen.
7. **Script 010-V2014 DiszMatcher setzen.sql** ausführen  
Es werden alle Disziplinen angezeigt.
8. **Script 015-V2009 Austragungsjahr setzen.sql** ausführen  
(In Fragefenster das Jahr des Anlasses angeben, 4-stellig )
9. **Script 020-V2012 Update DISZ VW J-A-FM-S.sql** ausführen  
Wenn noch Zeilen mit **NOT CHANGED** in der ersten Spalte des Resultates erscheinen, so muss die Option **Ausgabe editierbar** markiert werden und nach erneuter Ausführung in der Spalte des Feldes **DISZ\_MATCHER** eine Anpassung für **Rasen Kunststoff** oder **Wiese** eingerichtet werden. Beispiel **PS80\_A\_** wird zu **PS80\_K\_A\_** usw. Disziplinen die neu sind oder anlassspezifisch, werden im Status auf **NOT CHANGED** bleiben weil sie noch keine Wertung in Sportrator hinterlegt haben. Sie müssen später manuell konfiguriert werden. (Bitte solche Disziplinen notieren und später in der Grundkonfiguration korrekt definieren.)  
Am Ende sicherstellen, dass die Option **Ausgabe editierbar** wieder deaktiviert wird.
10. Wenn alter WV (wie bis 2009): **Script 023-V2009 Insert Fit\_and\_Fun\_Wertungsklassen Global.sql** ausführen. Wenn dieses Script eine Fehlermeldung erzeugt, so sind nicht alle Wertungsklassen importiert worden. Die erste nicht existierende Disziplin erzeugt eine Fehlermeldung und die anschliessenden werden nicht mehr importiert. Wird **beim ersten Durchlauf** eine Fehlermeldung generiert, so sind danach die folgenden Scripts einzeln und in der gegebenen Reihenfolge abzuarbeiten:  
**Script 024-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
**Script 025-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
**Script 026-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
**Script 027-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
**Script 028-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
**Script 029-V2009 Insert Fit\_and\_Fun\_Wertungsklassen.sql**  
 Jedes dieser Scripts kann einzeln fehlschlagen, gleichwohl müssen alle ausgeführt werden. Erst dann ist klar, welche Fit&Fun Wertungsklassen tatsächlich benötigt wurden.
11. **Script 038-V2011 Erzeuge Tabelle WERTUNGSKLASSE für alle restlichen Jugend Disziplinen.sql** ausführen.
12. Abfrageergebnis über Schaltfläche **Ablage** in eine Datei namens WERTUNGSKLASSE.txt sichern.

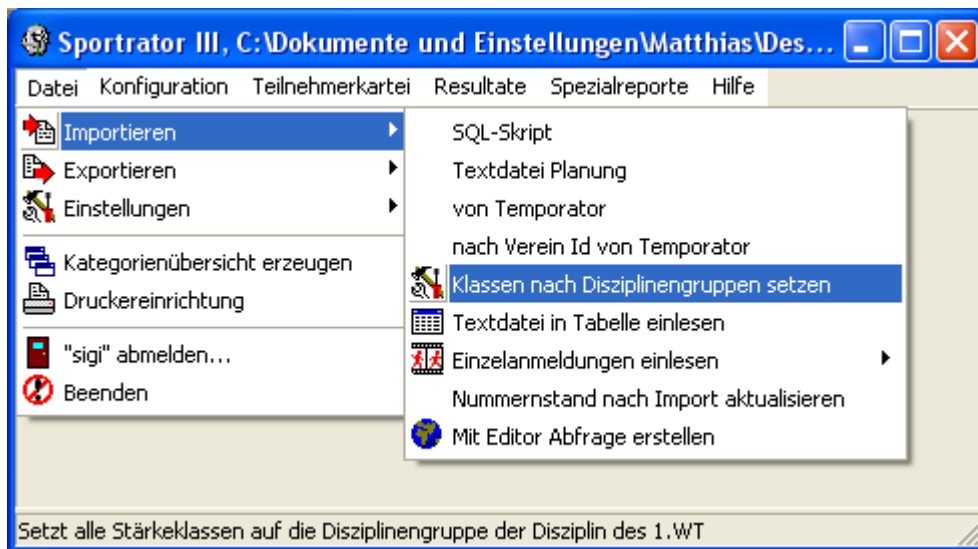


13. Zuvor gespeicherte Textdatei **WERTUNGSKLASSE.txt** nun über den Menüpunkt **Textdatei in Tabelle einlesen** importieren.



14. **Script 041-V2012\_Jugend\_Wertungsklassen update.sql** ODER (wenn alte WV) **Script 041-V2009\_Fit\_and\_Fun und Jugend\_Wertungsklassen update.sql** ausführen.  
In der Ausgabe sollten alle Zeilen mit einer VORGABE\_VARIANZ und einem VORGABE\_MITTEL befüllt sein (ausser es werden am Anlass noch echte Live-Auswertungen vorgenommen. In diesem Fall dürfen die den Live-Disziplinen zugehörigen Wertungsklassen in den Feldern VORGABE\_VARIANZ und VORGABE\_MITTEL leer bleiben. Bei Fixwertung deutet dies jedoch auf eine später manuell vorzunehmende Konfiguration hin).
15. (wenn alte WV) **Script 042-V2009 Fit\_and\_Fun\_Wertungsklassen update.sql** ausführen.

16. **Script 043-V2011 Fachtest update.sql** und **Script 041a-V2012 Jugend\_Wertungsklassen SPEZIAL update (alte Wertung in neuer Klasseneinteilung)** ausführen.
17. **Script 049-V2012 Jugend-Wertungsklassen Jahrgang update.sql** ausführen und im Abfragefenster das Austragungsjahr angeben. Mit dieser Abfrage werden alle Jahrgangsgrenzen der Wertungsklassen gesetzt.
18. **Script 050-V2009 Insert Disziplinengruppen.sql** ausführen. Mehrmaliges Ausführen wird fehlschlagen.
19. Textdatei **STKL\_DEF.txt** über den Menüpunkt **Textdatei in Tabelle einlesen** importieren. Dieses Script enthält alle vom STV definierten Stärkeklassen und für den spartenspezifischen Wettkampf alle Sparten, die in Sportrator auch als Stärkeklassen abgebildet werden. Wird eine Kategorie des VW am Anlass nicht ausgetragen, so werden die entsprechenden Stärkeklassen hier nicht importiert werden können. Daher müssen alle Fehlermeldungen hier mit **Ignorieren** quittiert werden.
20. **Script 055-V2009 Update Stärkeklassen Berechnungsart in Kat.sql** ausführen und Resultat kontrollieren
21. **Script 060-V2009 Update DG\_NR in DISZ.sql** ausführen und Resultat kontrollieren  
Die Spalte **DG\_NR** sollte überall einen Wert enthalten.
22. **Script 070-V2009 Update KEY\_NUM Basis nach Import.sql** ausführen.
23. **Klassen nach Disziplinengruppen setzen** bewirkt, dass die Stärkeklassen jeder Gruppe in den Kategorien mit spartenspezifischen Ranglisten (1-teilig) der ersten Disziplin entsprechend gesetzt wird. Dazu wird die Information im Feld **DG\_Kürzel** der **Disziplinengruppe** mit dem Feld **Kürzel** in der Definition der Stärkeklasse verglichen. Bei Übereinstimmung wird allen Gruppen einer Kategorie die Stärkeklasse (hier Sparte) zugeteilt, der die erste Disziplin der Gruppe angehört. Typischerweise wird dieses Kommando für alle Vereiswettkampf-Kategorien die einem 1-teiligen Wettkampf entsprechen ausgeführt, also z.B. für Kat. **112, 122, 132, 142, 152, 162, 172, 182**



24. In der Konfiguration muss **Klassenabdeckung aller Livedisziplinen überprüfen** ausgeführt werden. Nach drücken der Schaltfläche dürfen keine Fehler gemeldet werden.

**Konfiguration**

Benutzer | Veranstalter | Sparten | Auszeichnungen | Einzelkategorien | Einzeldisziplinen | Gruppenkategorien  
 Disziplinengruppen | Gruppendisziplinen | Gruppendisz. Liveauswertung | Aufgaben | Stärkekassen | Vereine | Histogramme

Histogrammklassen: 15 ☒ Histogramm darstellen ☒ Klassenabdeckung aller Livedisziplinen überprüfen

DG	Disz	Bezeichnung	Wertungsklasse	Geschl	von	bis	Anzahl V	Mittelw	Sigm	Klasse	Anzahl Vorg	Mittelwert für V	Sigma für V
Fit & FFF1	Fit im Te.	Frisbee Fra F	1900	900			7.5	8.64	Fix		0	7.5	2.5
Fit & FFF1	Fit im Te.	Step-Weit F	1901	901			52.727	1.545	Fix		0	7.5	2.5
Fit & FFF1	Fit im Te.	Frisbee Ma M	1900	900			8.25	4.25	Fix		0	7.5	2.5
Fit & FFF1	Fit im Te.	Step-Weit M	1901	901			58	27	Fix		0	7.5	2.5

0 0

☒ Vorschau

## 25. Script 75-V2009 Gruppen ohne Sparte und Stärkeklassenzuteilung suchen und anzeigen.sql ausführen.

Keine Zeile sollte in der letzten Spalte etwas anderes aufweisen als den Text **YES**

Falls alle Ausschreibungen nach STV erfolgten, sollte die Konfiguration für den Vereinswettkampf hier abgeschlossen sein. Spezialdisziplinen und offene Punkte müssen nun noch manuell konfiguriert werden.

## Ausführliche Tests einer Datenbank vor dem Anlass lohnen sich!

### Besonderes Augenmerk ist dabei folgenden Punkten zu schenken:

Die Stärkekassen aus der STV-Basiskonfiguration müssen praktisch bei jedem Anlass noch manuell den Wettkampfvorschriften des Organisators angeglichen werden.

Kategorien mit Disziplinen, für die vom Organisator eine **echte** Liveauswertung am Anlass vorgesehen ist, sollten nach Möglichkeit mit plausiblen Erfahrungswerten für Punktemittel und Sigma vorkonfiguriert werden. Dies erleichtert die Überprüfung der Konfiguration. Gute Erfahrungswerte dazu liegen bei einem Mittel von ca. 8.1 Punkten und einer Streuung von 0.9 Punkten um diesen Wert.

Anzahl Erfassungszeilen: 50

Ref. aus Leistungen

Referenz aus Punkten

Referenz für Livebewertung

Punktemittel: 8.1 Sigma: 0.9

Hauptgeschlecht

☐ Weiblich

☐ Männlich

☒ Keines

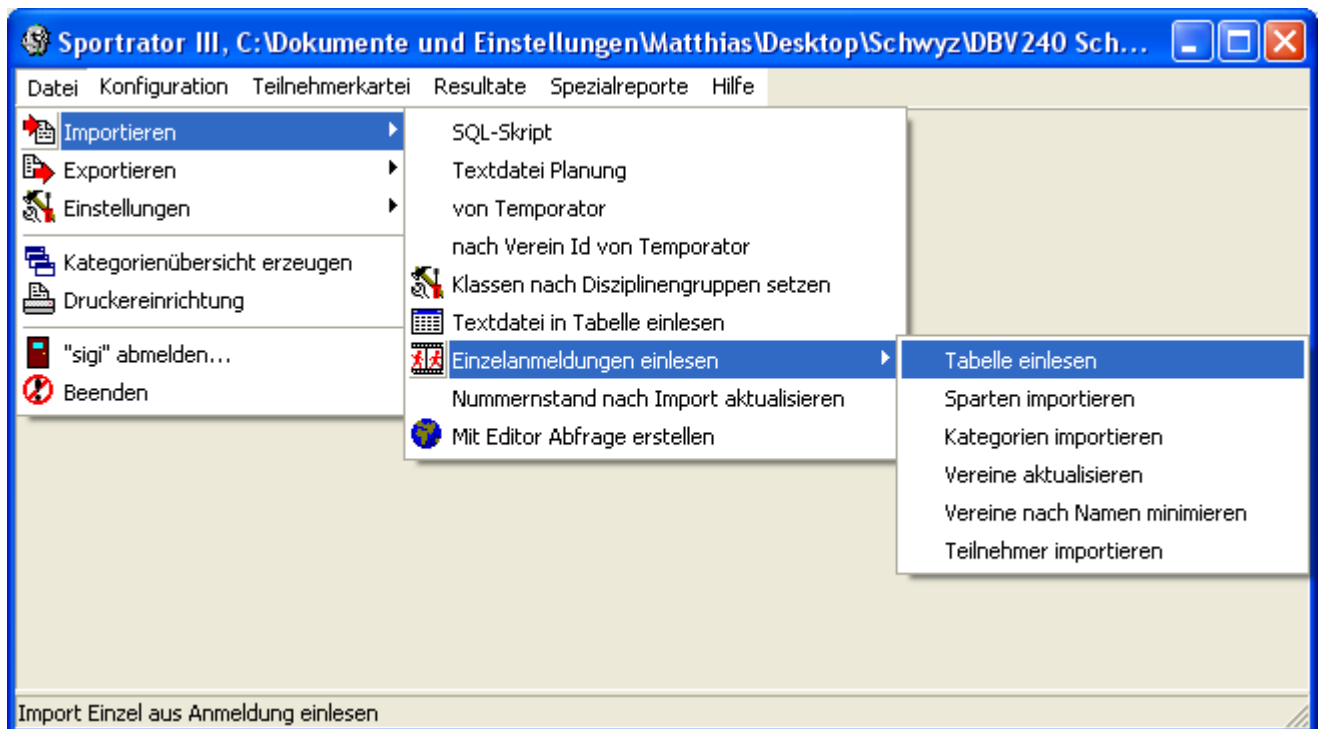
Sigma

☒ Stichprobe

☐ Population

## Einzelwettkampf konfigurieren

26. Vereinswettkampf wie beschrieben vorgängig konfigurieren
27. **Script 140-V2012 Insert Einzeldisziplinen analog Einheits-WV STV (INTERNET).sql** ausführen (Wenn Script fehlschlägt, erste Zeile im Script löschen und nochmals ausführen. (Abhängig von Vorgeschichte der Datenbank braucht es diese Zeile oder nicht)
28. **Script 141-V2012 Insert zusätzliche Einzeldisziplinen analog Einheits-WV STV (INTERNET).sql** ausführen
29. Falls es sich um eine früher bereits verwendete Datenbank handelt, kann es sein, dass die Tabelle Importeinzeln in der Datenbank zuwenig Felder aufweist zur Aufnahme des Imports aus dem Internet. Falls dies zutrifft, muss hier **Script 144-V2009 Importeinzeln Tabelle löschen zum neu erzeugen.sql** ausgeführt werden. **Danach werden die im Folgenden beschriebenen Schritte beim ersten Versuch fehlschlagen.** Dies ist aber bedeutungslos, da mit dem Aufruf der Funktion die Datenbanktabelle neu erzeugt wird. (Sie enthält danach sechs neue Felder für DISZ6 bis DISZ10). Neustarten der Applikation lässt das anschliessende Weiterarbeiten bei Schritt zu.
30. **IMPORTEINZEL.txt** aus Internetanmeldung importieren. Die Frage, ob eine Bestehende Tabelle gelöscht werden soll ist dabei mit Ja zu beantworten.



31. **Script 145-V2009 Vereine in Importeinzeln vorbehandeln.sql** ausführen. (Ausgabe der Daten kann ignoriert werden)
32. Importieren..Einzelmeldungen einlesen..**Sparten importieren** (Im Normalfall sind alle Sparten in Sportrator bereits vorhanden und werden somit nicht eingefügt. Eine anders lautende Meldung ist tendenziell verdächtig, ausser es handelt sich um einen Wettkampf mit neuen Anlansspezifischen Sparten.)
33. Importieren..Einzelmeldungen einlesen..**Kategorien importieren**
34. Importieren..Einzelmeldungen einlesen..**Vereine aktualisieren**
35. wenn Fehlermeldung im Schritt vorher, dann kann noch **Vereine nach Name minimieren** ausgeführt werden.
36. **Script 152-V2009 Liste mit fehlenden Einzelvereinen erzeugen für Tabelle VEREIN.sql** ausführen.



37. Resultat mit Schaltfläche **Ablage** als Datei unter dem Namen **VEREIN.txt** speichern.
38. Soeben erzeugte Datei **VEREIN.txt** mit Schaltfläche **Textdatei in Tabelle Einlesen** wieder importieren
39. Importieren..Einzelanmeldungen einlesen..**Vereine aktualisieren**
40. wenn Fehlermeldung im Schritt vorher, dann kann noch **Vereine nach Name minimieren** ausgeführt werden. (nur Präfix Ort und Kanton müssen hier stimmen, damit der Verein erkannt werden kann)
41. **Script 152-V2009 Liste mit fehlenden Einzelvereinen erzeugen für Tabelle VEREIN.sql** erneut ausführen. Nun sollte das Resultat leer sein. Damit können nun die Teilnehmer importiert werden.
42. **Script 154-V2009 Manipulation der ungültigen Jahrgänge in Einzelimport.sql** ausführen. Dieses Script ermöglicht auch das Filtern auf den Daten der Tabelle Importeinzeln. Diese Zwischentabelle enthält die Rohdaten aus der Datei Importeinzeln.txt. ist aber viel toleranter als die Personentabelle, in der die Teilnehmer im nächsten Schritt eingelesen werden. Daher kann in Importeinzeln noch einiges stehen, was im nächsten Schritt nicht importiert werden kann und daher erst manuell angepasst werden muss.
43. **Sportrator verlassen und neu starten.**
44. Importieren..Einzelanmeldungen einlesen..**Teilnehmer Importieren**
45. Importieren..**Nummernstand nach Import aktualisieren**
46. **Script 158-V2009 Zeige allenfalls nicht importierte Personen aus Importeinzeln.sql** ausführen. Diese Liste sollte leer sein. Wird die Option **Ausgabe editierbar** aktiviert, so können die Fehler in der Ausgabe direkt korrigiert werden. Script 158... ist dazu aber nicht direkt fähig. Zeilen notieren und Script 154... in diesem Fall nochmals verwenden. (Evtl. **Augenmerk auf Spalte JAHRGANG1**, dort muss ein vierstelliger Wert > 1900 stehen, wenn das Feld nicht leer ist.)
47. **Script 170-V2009 Update KEY\_NUM Basis nach Import.sql** ausführen
48. **Script 180-V2009 Unvollständig definierte Einzelkategorien anzeigen.sql** ausführen  
Hier wird ersichtlich, was noch zu definieren bleibt.

Mit **VORSICHT** kann am Ende, nach allen Tests und Konfigurationsarbeiten, wenn fest steht, dass keine Teilnehmer mehr in bisher leer gebliebenen Kategorien angemeldet werden müssen, das **Script 190-V2009 Leere und unbenutzte Kategorien löschen.sql** ausgeführt werden.

### Ausführliche Tests einer Datenbank vor dem Anlass lohnen sich!

**Besonderes Augenmerk ist dabei folgenden Punkten zu schenken:**

- Beim Test des Einzelwettkampfes ist besonderes darauf zu achten, dass alle Notenblätter mit der Reihenfolge der Disziplinen in den entsprechenden Erfassungsmasken korrespondieren.
- Die Reihenfolge der Disziplinen kann in den Aufgaben der jeweiligen Kategorie festgelegt und verändert werden. Bei den Spalten **Pos. in Form** und **Pos. auf Report** ist zwingend auf Eindeutigkeit der angegebenen Werte zu achten. Gültige Werte für den Report liegen im Bereich zwischen 1 und 20

**Abfrage**

Tabellen: GEMEINDE, GRUPPE, **IMPORTEINZEL**, KAT, KEY\_NUM, LEGAT\_SOURCE, LEISTUNG

Felder: KAT\_NR, START\_NR, SPARTE\_BEZ, KAT\_BEZ, PREFIX, ORT, VEREIN\_ID

SQL-Anweisung: Z: 11, S: 26, ☐ Zeilenumbruch

```
select distinct
i.KAT_NR as KAT_NR,
i.kat_bez,
i.SPARTE_BEZ as sparte,
i.KAT_BEZ as bez_import,
K.KAT_BEZ as bez_sportrator
from importeinzel i, KAT K where
i.kat_nr = K.kat_nr
and
i.KAT_bez <> k.kat_bez;
```

Ausgabe: Zeilen: 11, Position: -1

Result | Updater

Update Table: IMPORTEINZEL

Values:

- ☐ KAT\_NR
- ☒ **KAT\_BEZ**
- ☐ SPARTE
- ☐ BEZ\_IMPORT
- ☐ BEZ\_SPORTRATOR

Key Fields:

- ☒ **KAT\_NR**
- ☐ KAT\_BEZ
- ☐ SPARTE
- ☐ BEZ\_IMPORT
- ☐ BEZ\_SPORTRATOR

Conditions:

☐ Ausgabe editierbar