Aufgabe 20
1)
Termgewichte W(t,d) = tf(t,d) / df(t)

Ti1 = Anwendung von Tenormin bei Krankeiten des Herzens
Ti2 = Zusammenhang zwischen Beschwerden der Lunge und Beschwerden des Herzens
Ti3 = Tenormin bei Lungenkrankheiten

die Vorkommenshäufigkeiten von Wörter (tf (t,d))

	Ti 1	Ti 2	Ti 3
Anwendung	1	0	0
Herz	1	1	0
Tenormin	1	0	1
Krankheit	1	0	0
Zusammenhang	0	1	0
Beschwerde	0	2	0
Lunge	0	1	0
Lungenkrankheit	0	0	1

Term	Anzahl der Doc(df (t))
Anwendung	3000
Herz	500
Tenormin	40
Krankheit	4000
Zusammenhang	1500
Beschwerde	2000
Lunge	600
Lungenkrankheit	200

Die Termgewichte der Indexterme:

	Ti 1	Ti 2	Ti 3
Anwendung	0.000333	0	0
Herz	0.002	0.002	0
Tenormin	0.025	0	0.025
Krankheit	0.00025	0	0
Zusammenhang	0	0.000667	0
Beschwerde	0	0.001	0
Lunge	0	0.001667	0
Lungenkrankheit	0	0	0.005

2)
Um die Relevanz zu berechnen werden die Anfragen und Dokumente im Vektorraum übersetzt und verglichen. Je ähnlicher sich Anfrage- und Dokumentvektor sind, desto höher wird die Relevanz des jeweiligen Dokuments eingeschätzt. Die Ähnlichkeit gibt somit auch vor, wie das Ranking zu laufen hat.
Um die Ähnlichkeit zwischen Anfragen und Dokumente zu berechnen ergibt sich Cosinus-Maß, die effektiv den Cosinus des Winkels zwischen den beiden Vektoren berechnet.

$$\vec{v}_1 \cdot \vec{v}_2 = |\vec{v}_1||\vec{v}_2| \cos(\vec{v}_1, \vec{v}_2) \Leftrightarrow \cos(\vec{v}_1, \vec{v}_2) = \frac{\vec{v}_1 \cdot \vec{v}_2}{|\vec{v}_1||\vec{v}_2|}$$

In der Anfrage: "Tenormin für Beschwerden des Herzens" treten die Wörter „Tenormin“, „Beschwerde“ und „Herz“ jeweils einmal auf. Deshalb: W(q, Tenormin) = 1;
W(q, Beschwerde) = 1 ; W(q, Herz) = 1. Beim Cosinus-Maß kann der Beitrag der Anfrage-Länge zum Nenner igoniert werden:
cos(q, d) = sum(W(q,t)*W(d,t))/ sqrt(sum(W(d,t)*W(d*t)))
{q für Query, t für Term, d für Dokument}

cos(q, Ti1) = (1*0.025+1*0+1*0.002)/sqrt((0.002*0.002) + (0.025*0.025)) = 1.0766
cos(q, Ti2) = (1*0 + 1 *0.001 + 1*0.002)/ sqrt((0.001 *0.001 + 0.002*0.002)) = 1.342
cos(q, Ti3) = 1*0.025 /sqrt(0.025*0.025) = 1
Nach Relevanz ist Ranking von Dokumente: Ti2, Ti1, Ti3

2) Nach der Methode „exact match“ wird kein Treffer gefunden, weil es kein Dokument gibt, in dem alle drei Terme enthalten.

Aufgabe 21

Termgewichte W(t,d) = tf(t,d) * idf(t)

Term	Doc	TF	IDF	TF-IDF
car	doc1	27	1,65	44,55
	doc2	4	1,65	6,6
	doc3	24	1,65	39,6
auto	doc1	3	2,08	6,24
	doc2	33	2,08	68,64
	doc3	0	2,08	0
insurance	doc1	0	1,62	0
	doc2	33	1,62	53,46
	doc3	29	1,62	46,98
best	doc1	14	1,5	21
	doc2	0	1,5	0
	doc3	17	1,5	25,5

Aufgabe22

Termgewichte W(t,d) = tf(t,d) * idf(t) = tf(t,d) * log N/df(t) (Hier: log Basis 2)

tf(w,D)	D1	D2	D3	D4	D5	D6	D7	D8	D9	d10
w1	3	0	0	5	12	0	0	2	8	1
w2	8	6	0	12	0	0	9	1	3	10
w3	0	1	7	0	1	5	12	0	2	0

N = 10
df(w1) = 6; df(w2)=7; df(w3) = 6

der tf-idf-Wert von w1 in:
d1: w11 = 3 * log 10/6 = 2.21
d4: w14 = 5* log 10/6 = 3.68
d5: w15 = 12* log 10/6 = 8.84
d8: w18 = 2* log10/6 = 1.47
d9: w19 = 8 * log 10/6 = 5.90
d10: w 110= 1 * log 10/6 = 0.74

der tf-idf-Wert von w2 in:
d1: w21 = 8 * log 10/7 = 4.12
d2: w22 = 6 * log 10/7 = 3.09
d4: w24 = 12 * log 10/7 = 6.17
d7: w27 = 9 * log 10/7 = 4.63
d8: w28 = 1 * log 10/7 = 0.51
d9: w29 = 3 * log 10/7 = 1.54
d10: w210 = 10 * log 10/7 = 5.15

der tf-idf-Wert von w3 in:
d2: w32 = 1 * log 10/6= 0.74
d3: w33 = 7 * log 10/6 = 5.16
d5: w35 = 1 * log 10/6 = 0.74
d6: w36 = 5 * log 10/6 = 3.68
d7: w37 = 12 * log 10/6 = 8.84
d9: w39 = 2 * log 10/6 = 1.47