

## Session 2: Key-Value Stores (kompakt)

**Lernziele:** Key-Value Paradigma verstehen | Zugriffsmuster & Patterns kennen | Grenzen & Trade-offs bewerten

### Rückblick: Was haben wir beim letzten Mal gelernt?

Willkommen zurück! Bevor wir in Key-Value Stores eintauchen, lasst uns kurz rekapitulieren, wo wir letzte Woche geendet haben.

#### Session 1 Recap

- **Rohdaten-Formate:** CSV, JSON, YAML, XML
- **Ad-hoc Analysen:** Programmiersprachen direkt auf CSV
- **Problem erkannt:** Lineare Suche =  $O(n)$  Zeitkomplexität
- **Inkonsistenzen:** Keine Validierung, keine Integrität

Wir haben gesehen, dass Dateien flexibel sind, aber bei strukturierten Zugriffen schnell an Grenzen stoßen. Heute schauen wir uns an, wie Key-Value Stores dieses Problem lösen.

**Zentrale Frage aus:** Wie können wir schneller als  $O(n)$  auf Daten zugreifen?

Wie dargestellt, benötigen wir linearen Zeitaufwand, um in eine Datei nach einem Eintrag zu suchen zu durchsuchen. Wenn ihr noch nicht wisst, was  $O(n)$  bedeutet, keine Sorge – wir werden das gleich im Detail besprechen!

### Das Problem: CSV-Suche ist langsam

Nehmen wir eine Produktdatenbank als Beispiel und schauen wir uns an, was passiert, wenn wir ein einzelnes Produkt suchen müssen.

**Szenario:** Wir haben eine [CSV-Datei mit Produkten](#) und wollen Details zu Produkt **P00567** finden.

product_id	name	category	brand	price
P00001	Laptop 14" 2.0	Electronics	PixelPeak	1399.03
P00002	Portable SSD Max M	Electronics	ZenCore	805.93
P00003	4K Monitor - Green	Electronics	Neutrino	522.48
P00004	Tablet 10"	Electronics	Voltix	985.75
P00005	Portable SSD Lite	Electronics	OrbiTech	508.23
P00006	Soundbar - Red	Electronics	OrbiTech	786.98
P00007	Mechanical Keyboard - Black	Electronics	Neutrino	537.59
P00008	Laptop 14" Plus	Electronics	Voltix	1187.46
P00009	Action Camera - Green	Electronics	OrbiTech	502.12
P00010	Soundbar	Electronics	Auralite	1340.90
P00011	4K Monitor	Electronics	Neutrino	24.92
P00012	Action Camera XL	Electronics	Auralite	636.73
P00013	Gaming Mouse Plus M	Electronics	Voltix	787.02
P00014	Bluetooth Speaker Plus	Electronics	Auralite	865.42
	Noise-			

Product ID	Product Name	Category	Supplier	Unit Price
P00015	Cancelling Earbuds Lite	Electronics	Auralite	1339.39
P00016	Action Camera - Blue	Electronics	ZenCore	1322.90
P00017	Laptop 14" Pro S	Electronics	Auralite	481.53
P00018	Gaming Mouse	Electronics	Neutrino	1402.13
P00019	USB-C Hub - White	Electronics	BlueWave	1093.57
P00020	USB-C Hub - Red	Electronics	PixelPeak	317.33
P00021	Wireless Headphones	Electronics	BlueWave	377.20
P00022	Laptop 14" 2.0 - Black	Electronics	Neutrino	833.21
P00023	Wireless Headphones M	Electronics	BlueWave	697.05
P00024	Tablet 10" - Green	Electronics	Auralite	667.70
P00025	Soundbar Plus	Electronics	PixelPeak	739.53
P00026	USB-C Hub Plus	Electronics	Auralite	1356.47
P00027	Noise-Canceling Earbuds Max	Electronics	PixelPeak	1352.11
P00028	Tablet 10" - Black	Electronics	Neutrino	128.34

Global Product Portfolio Analysis - Q3 2024				
Product ID	Product Name	Category	Supplier	Unit Price
P00029	Mechanical Keyboard - Blue	Electronics	Voltix	1106.59
P00030	Action Camera	Electronics	ZenCore	1027.70
P00031	Tablet 10" Plus	Electronics	PixelPeak	648.45
P00032	Noise-Canceling Earbuds	Electronics	Auralite	784.44
P00033	Smartwatch - White	Electronics	ZenCore	61.61
P00034	Bluetooth Speaker M	Electronics	Neutrino	853.91
P00035	4K Monitor	Electronics	Neutrino	62.90
P00036	Tablet 10"	Electronics	ZenCore	852.45
P00037	Action Camera Plus	Electronics	Auralite	1376.90
P00038	Bluetooth Speaker Pro	Electronics	OrbiTech	725.66
P00039	Portable SSD Lite S	Electronics	OrbiTech	546.80
P00040	Gaming Mouse - Graphite	Electronics	Auralite	1297.21
P00041	Portable SSD Lite - Graphite	Electronics	OrbiTech	1242.64
P00042	Wireless Headphones	Electronics	OrbiTech	523.27

	Pro S - Silver			
P00043	Gaming Mouse M	Electronics	PixelPeak	1175.69
P00044	Bluetooth Speaker Lite M	Electronics	Voltix	1425.09
P00045	Gaming Mouse Lite S	Electronics	Auralite	486.97
P00046	4K Monitor - White	Electronics	BlueWave	1390.69
P00047	4K Monitor	Electronics	OrbiTech	226.26
P00048	4K Monitor	Electronics	Voltix	748.05
P00049	Bluetooth Speaker XL	Electronics	OrbiTech	215.56
P00050	Gaming Mouse Pro S	Electronics	BlueWave	270.97
P00051	Smartwatch Pro L - Graphite	Electronics	Voltix	394.22
P00052	Portable SSD Plus	Electronics	OrbiTech	271.37
P00053	Tablet 10"	Electronics	Voltix	1433.70
P00054	4K Monitor	Electronics	Auralite	962.86
P00055	USB-C Hub	Electronics	Auralite	846.15
P00056	Wireless Headphones Max	Electronics	Auralite	639.32
P00057	USB-C Hub Pro	Electronics	PixelPeak	1371.22

P00058	4K Monitor	Electronics	Neutrino	1368.99
P00059	Action Camera 2.0	Electronics	Voltix	885.27
P00060	Mechanical Keyboard	Electronics	Voltix	1462.40
P00061	Action Camera	Electronics	OrbiTech	1259.30
P00062	Laptop 14" 2.0 - White	Electronics	Voltix	132.35
P00063	Portable SSD	Electronics	Voltix	289.75
P00064	USB-C Hub S	Electronics	PixelPeak	298.11
P00065	Gaming Mouse Plus S	Electronics	BlueWave	562.30
P00066	Soundbar Plus	Electronics	ZenCore	146.53
P00067	Smartwatch Pro - Blue	Electronics	BlueWave	631.64
P00068	4K Monitor L	Electronics	Voltix	812.89
P00069	Gaming Mouse M	Electronics	BlueWave	1042.31
P00070	Smartwatch S	Electronics	Voltix	870.07
P00071	Soundbar Pro S	Electronics	OrbiTech	1425.74
P00072	Soundbar 2.0	Electronics	ZenCore	231.87
P00073	Soundbar Max M	Electronics	Neutrino	317.14
P00074	USB-C Hub -	Electronics	ZenCore	1337.23

Product ID	Color	Category	Supplier	Unit Price
P00074	Black	Electronics	ZenCore	1001.20
P00075	4K Monitor	Electronics	OrbiTech	1445.86
P00076	USB-C Hub XL	Electronics	BlueWave	506.06
P00077	Tablet 10" S - Black	Electronics	ZenCore	192.98
P00078	Mechanical Keyboard 2.0 XL	Electronics	OrbiTech	1151.32
P00079	Smartwatch	Electronics	Auralite	1203.52
P00080	Bluetooth Speaker	Electronics	Voltix	1097.74
P00081	Wireless Headphones	Electronics	Voltix	852.73
P00082	Gaming Mouse	Electronics	ZenCore	317.38
P00083	Portable SSD 2.0	Electronics	Neutrino	1131.97
P00084	4K Monitor XL	Electronics	ZenCore	1389.73
P00085	Portable SSD	Electronics	Neutrino	968.47
P00086	Tablet 10" Pro - Black	Electronics	Voltix	827.84
P00087	Laptop 14"	Electronics	OrbiTech	772.23
P00088	Laptop 14" Lite	Electronics	PixelPeak	1290.81
P00089	Tablet 10" - Graphite	Electronics	Voltix	1101.99
	Tablet 10"			

P00090	Speaker Max	Electronics	Neutrino	1019.90
P00091	Gaming Mouse Pro	Electronics	Auralite	415.47
P00092	Smartwatch	Electronics	ZenCore	1145.45
P00093	Bluetooth Speaker - Blue	Electronics	ZenCore	636.58
P00094	Wireless Headphones	Electronics	Auralite	995.22
P00095	Gaming Mouse	Electronics	ZenCore	92.53
P00096	Bluetooth Speaker	Electronics	ZenCore	1493.82
P00097	Action Camera - White	Electronics	ZenCore	129.17
P00098	Soundbar Max	Electronics	Neutrino	1145.22
P00099	USB-C Hub Plus	Electronics	Voltix	408.92
P00100	USB-C Hub S	Electronics	Voltix	706.74
P00101	Leggings Max - Silver	Clothing	AeroLoom	245.35
P00102	Jacket Max XL	Clothing	Cotton&Co	110.77
P00103	Socks (5-pack) 2.0 M	Clothing	Nordline	84.42
P00104	Hoodie - Black	Clothing	UrbanTrail	106.25

P00105	Basic T-Shirt Max	Clothing	AeroLoom	24.34
P00106	Hoodie	Clothing	Nordline	184.13
P00107	Jacket	Clothing	UrbanTrail	44.00
P00108	Raincoat 2.0	Clothing	UrbanTrail	131.54
P00109	Jeans XL	Clothing	AeroLoom	113.52
P00110	Jeans	Clothing	UrbanTrail	104.87
P00111	Chinos Pro - Red	Clothing	Nordline	195.07
P00112	Polo Shirt	Clothing	AeroLoom	121.06
P00113	Jacket L	Clothing	AeroLoom	29.28
P00114	Raincoat	Clothing	AeroLoom	43.81
P00115	Jeans - Silver	Clothing	PureThread	34.16
P00116	Jacket	Clothing	AeroLoom	140.53
P00117	Leggings L	Clothing	Seaside	18.10
P00118	Jacket	Clothing	PureThread	223.66
P00119	Sneakers Pro XL	Clothing	Meridian Wear	185.79
P00120	Basic T-Shirt Lite	Clothing	PureThread	75.34
P00121	Hoodie M	Clothing	PureThread	52.36
P00122	Jeans	Clothing	Cotton&Co	35.78
P00123	Polo Shirt 2.0	Clothing	AeroLoom	235.50
P00124	Basic T-Shirt 2.0 - Graphite	Clothing	PureThread	51.30

P00125	Chinos L - Silver	Clothing	Meridian Wear	205.43
P00126	Hoodie - Green	Clothing	Nordline	237.98
P00127	Dress Shirt - White	Clothing	Seaside	91.62
P00128	Chinos	Clothing	Seaside	138.48
P00129	Jeans	Clothing	Nordline	73.34
P00130	Raincoat	Clothing	PureThread	143.43
P00131	Cap - Black	Clothing	AeroLoom	205.26
P00132	Dress Shirt Plus - Graphite	Clothing	PureThread	198.67
P00133	Socks (5-pack)	Clothing	Seaside	159.64
P00134	Polo Shirt - Red	Clothing	Meridian Wear	243.71
P00135	Sneakers 2.0	Clothing	UrbanTrail	81.46
P00136	Jacket	Clothing	Seaside	53.04
P00137	Hoodie	Clothing	Cotton&Co	108.49
P00138	Hoodie Plus L	Clothing	Cotton&Co	129.18
P00139	Jacket 2.0	Clothing	Seaside	143.66
P00140	Cap Lite	Clothing	Nordline	35.55
P00141	Dress Shirt - Blue	Clothing	AeroLoom	39.97
P00142	Socks (5-pack)	Clothing	PureThread	116.91

P00143	Socks (5-pack) Max S	Clothing	Meridian Wear	127.88
P00144	Jeans	Clothing	Nordline	141.04
P00145	Cap - Green	Clothing	Nordline	193.15
P00146	Jacket - Green	Clothing	Nordline	122.70
P00147	Jeans	Clothing	Cotton&Co	54.21
P00148	Chinos	Clothing	Cotton&Co	8.64
P00149	Basic T-Shirt	Clothing	Seaside	102.40
P00150	Leggings L	Clothing	AeroLoom	11.57
P00151	Polo Shirt - Silver	Clothing	PureThread	227.75
P00152	Leggings Max S	Clothing	Nordline	233.50
P00153	Raincoat	Clothing	Nordline	94.33
P00154	Jeans Plus L	Clothing	Meridian Wear	223.62
P00155	Basic T-Shirt Lite	Clothing	UrbanTrail	146.12
P00156	Dress Shirt Max	Clothing	Cotton&Co	151.34
P00157	Polo Shirt Pro XL	Clothing	Cotton&Co	14.93
P00158	Cap Pro	Clothing	Nordline	71.08
P00159	Basic T-Shirt XL	Clothing	AeroLoom	83.32
P00160	Leggings Lite	Clothing	PureThread	18.50

P00161	Dress Shirt	Clothing	Meridian Wear	24.95
P00162	Socks (5-pack)	Clothing	Cotton&Co	237.02
P00163	Raincoat	Clothing	AeroLoom	70.25
P00164	Sneakers - Blue	Clothing	Nordline	228.06
P00165	Leggings XL	Clothing	AeroLoom	180.85
P00166	Cap 2.0	Clothing	AeroLoom	220.22
P00167	Leggings - Black	Clothing	PureThread	47.96
P00168	Leggings 2.0 - Green	Clothing	Cotton&Co	222.35
P00169	Socks (5-pack) Max	Clothing	PureThread	82.45
P00170	Sneakers	Clothing	UrbanTrail	9.79
P00171	Leggings	Clothing	Nordline	31.28
P00172	Raincoat Pro	Clothing	Nordline	103.93
P00173	Chinos	Clothing	PureThread	214.42
P00174	Hoodie 2.0 XL - Blue	Clothing	PureThread	152.38
P00175	Hoodie Lite L - White	Clothing	Nordline	30.03
P00176	Jeans Max	Clothing	AeroLoom	12.52
P00177	Polo Shirt	Clothing	AeroLoom	61.45
P00178	Jacket	Clothing	AeroLoom	155.19
P00179	Leggings Plus	Clothing	AeroLoom	82.65

	L			
P00180	Chinos	Clothing	Cotton&Co	135.28
P00181	Dress Shirt	Clothing	PureThread	103.60
P00182	Dress Shirt 2.0 S	Clothing	Seaside	6.13
P00183	Jeans 2.0 - Blue	Clothing	Nordline	164.43
P00184	Hoodie - Black	Clothing	Nordline	196.66
P00185	Sneakers	Clothing	Nordline	51.40
P00186	Chinos Lite	Clothing	Seaside	126.13
P00187	Chinos	Clothing	Meridian Wear	52.96
P00188	Hoodie	Clothing	UrbanTrail	204.27
P00189	Polo Shirt	Clothing	Meridian Wear	240.17
P00190	Basic T-Shirt Lite S	Clothing	UrbanTrail	63.72
P00191	Jeans Pro - Graphite	Clothing	Nordline	171.82
P00192	Jacket Lite L - Red	Clothing	Seaside	222.71
P00193	Hoodie	Clothing	UrbanTrail	34.90
P00194	Sneakers L	Clothing	Meridian Wear	177.36
P00195	Dress Shirt Pro	Clothing	UrbanTrail	198.34
P00196	Jeans	Clothing	Nordline	231.74
P00197	Polo Shirt S	Clothing	AeroLoom	85.13

P00198	Leggings	Clothing	Nordline	40.05
P00199	Chinos Max M	Clothing	PureThread	38.03
P00200	Jacket Plus M	Clothing	Meridian Wear	40.52
P00201	Granola 750g	Groceries	GreenVale	31.60
P00202	Olive Oil 1L - Silver	Groceries	Everfarm	26.73
P00203	Tomato Sauce 400g Max XL	Groceries	Everfarm	34.12
P00204	Olive Oil 1L M	Groceries	DailyChoice	36.10
P00205	Green Tea 50 bags M	Groceries	FreshField	26.45
P00206	Green Tea 50 bags - Black	Groceries	SunHarvest	20.04
P00207	Arabica Coffee Beans 1kg	Groceries	FreshField	0.90
P00208	Pasta Fusilli 500g Plus	Groceries	Everfarm	38.17
P00209	Olive Oil 1L M - Green	Groceries	GreenVale	18.64
P00210	Green Tea 50 bags Max	Groceries	Everfarm	35.94
P00211	Arabica Coffee Beans 1kg Plus	Groceries	FreshField	19.88
P00212	Sea Salt 500g	Groceries	Everfarm	19.40
P00213	Green Tea 50 bags	Groceries	Pure&Simple	38.52

P00214	Arabica Coffee Beans 1kg	Groceries	SunHarvest	37.91
P00215	Pasta Fusilli 500g Max M	Groceries	GreenVale	4.49
P00216	Basmati Rice 1kg L - White	Groceries	DailyChoice	18.39
P00217	Tomato Sauce 400g	Groceries	FreshField	10.35
P00218	Dark Chocolate 85% 100g XL - Green	Groceries	Everfarm	30.33
P00219	Tomato Sauce 400g	Groceries	GreenVale	29.10
P00220	Basmati Rice 1kg Plus - White	Groceries	FreshField	33.79
P00221	Basmati Rice 1kg Max L	Groceries	SunHarvest	4.76
P00222	Basmati Rice 1kg Lite S	Groceries	DailyChoice	38.30
P00223	Granola 750g Plus - Green	Groceries	Everfarm	16.09
P00224	Organic Oat Milk 1L	Groceries	DailyChoice	19.23
P00225	Dark Chocolate 85% 100g	Groceries	DailyChoice	37.93
P00226	Sea Salt 500g Max - Blue	Groceries	DailyChoice	29.72

P00227	Tomato Sauce 400g - Black	Groceries	Pure&Simple	5.72
P00228	Olive Oil 1L	Groceries	FreshField	29.25
P00229	Organic Oat Milk 1L	Groceries	SunHarvest	6.03
P00230	Green Tea 50 bags - Silver	Groceries	Everfarm	23.99
P00231	Honey 500g 2.0 - White	Groceries	SunHarvest	34.76
P00232	Sea Salt 500g - White	Groceries	Everfarm	34.37
P00233	Olive Oil 1L	Groceries	Pure&Simple	34.54
P00234	Green Tea 50 bags S - Red	Groceries	Everfarm	8.08
P00235	Organic Oat Milk 1L	Groceries	GreenVale	26.05
P00236	Arabica Coffee Beans 1kg Pro XL - Blue	Groceries	DailyChoice	33.57
P00237	Honey 500g 2.0 M	Groceries	DailyChoice	13.14
P00238	Olive Oil 1L M	Groceries	GreenVale	15.43
P00239	Arabica Coffee Beans 1kg Lite	Groceries	Pure&Simple	21.66
P00240	Granola 750g Max	Groceries	FreshField	6.11
P00241	Basmati Rice	-	-	-

P00241	1kg Pro	Groceries	Evertarm	4.39
P00242	Almonds 200g Lite XL	Groceries	Everfarm	2.48
P00243	Pasta Fusilli 500g Max XL	Groceries	FreshField	7.20
P00244	Basmati Rice 1kg	Groceries	Pure&Simple	14.49
P00245	Organic Oat Milk 1L	Groceries	GreenVale	13.61
P00246	Sea Salt 500g	Groceries	FreshField	38.41
P00247	Dark Chocolate 85% 100g Lite L - Green	Groceries	DailyChoice	23.61
P00248	Pasta Fusilli 500g Lite M	Groceries	DailyChoice	13.27
P00249	Sea Salt 500g Lite - Graphite	Groceries	FreshField	3.07
P00250	Organic Oat Milk 1L Lite S - Red	Groceries	SunHarvest	2.25
P00251	Honey 500g Lite S	Groceries	Everfarm	8.13
P00252	Tomato Sauce 400g	Groceries	FreshField	23.40
P00253	Basmati Rice 1kg 2.0 - Red	Groceries	Pure&Simple	5.32
P00254	Green Tea 50 bags Plus	Groceries	GreenVale	39.84

P00255	Almonds 200g - Green	Groceries	FreshField	31.09
P00256	Granola 750g XL - Graphite	Groceries	Everfarm	1.99
P00257	Green Tea 50 bags	Groceries	Everfarm	29.26
P00258	Tomato Sauce 400g - White	Groceries	Pure&Simple	27.36
P00259	Pasta Fusilli 500g - Black	Groceries	Everfarm	32.26
P00260	Arabica Coffee Beans 1kg - Green	Groceries	FreshField	18.68
P00261	Sea Salt 500g - Green	Groceries	DailyChoice	13.10
P00262	Green Tea 50 bags XL	Groceries	GreenVale	15.17
P00263	Honey 500g - Black	Groceries	Pure&Simple	1.73
P00264	Almonds 200g	Groceries	DailyChoice	3.59
P00265	Granola 750g - White	Groceries	DailyChoice	30.38
P00266	Dark Chocolate 85% 100g - White	Groceries	Pure&Simple	15.75
P00267	Dark Chocolate 85% 100g - White	Groceries	SunHarvest	22.69

P00268	Pasta Fusilli 500g - Silver	Groceries	DailyChoice	38.97
P00269	Tomato Sauce 400g L - Black	Groceries	GreenVale	16.03
P00270	Sea Salt 500g	Groceries	SunHarvest	26.67
P00271	Almonds 200g Max	Groceries	Everfarm	26.28
P00272	Olive Oil 1L - Silver	Groceries	DailyChoice	5.41
P00273	Tomato Sauce 400g Max	Groceries	GreenVale	10.80
P00274	Almonds 200g - Graphite	Groceries	DailyChoice	28.17
P00275	Granola 750g Plus	Groceries	SunHarvest	32.53
P00276	Granola 750g	Groceries	Pure&Simple	39.66
P00277	Almonds 200g Max - White	Groceries	Everfarm	7.20
P00278	Sea Salt 500g - White	Groceries	Pure&Simple	24.16
P00279	Honey 500g	Groceries	DailyChoice	13.03
P00280	Arabica Coffee Beans 1kg	Groceries	SunHarvest	3.09
P00281	Honey 500g Max	Groceries	DailyChoice	34.67

P00282	Arabica Coffee Beans 1kg	Groceries	SunHarvest	36.58
P00283	Dark Chocolate 85% 100g L - Red	Groceries	FreshField	20.34
P00284	Organic Oat Milk 1L Max	Groceries	FreshField	5.33
P00285	Green Tea 50 bags	Groceries	Pure&Simple	3.68
P00286	Green Tea 50 bags	Groceries	SunHarvest	32.88
P00287	Sea Salt 500g	Groceries	Everfarm	16.29
P00288	Organic Oat Milk 1L 2.0	Groceries	Pure&Simple	6.86
P00289	Green Tea 50 bags	Groceries	SunHarvest	38.07
P00290	Honey 500g Max S	Groceries	DailyChoice	36.37
P00291	Honey 500g - Black	Groceries	SunHarvest	28.21
P00292	Basmati Rice 1kg	Groceries	GreenVale	2.33
P00293	Granola 750g 2.0 - Blue	Groceries	DailyChoice	37.53
P00294	Dark Chocolate 85% 100g M - Black	Groceries	GreenVale	28.28
	Sea Salt 500g			

P00295	Max XL - Silver	Groceries	DailyChoice	37.43
P00296	Green Tea 50 bags - Black	Groceries	FreshField	17.03
P00297	Arabica Coffee Beans 1kg Plus - Blue	Groceries	DailyChoice	17.32
P00298	Organic Oat Milk 1L Plus M	Groceries	DailyChoice	15.99
P00299	Arabica Coffee Beans 1kg Max XL - Silver	Groceries	DailyChoice	22.38
P00300	Dark Chocolate 85% 100g Max S - Silver	Groceries	SunHarvest	10.67
P00301	Wireless Mouse XL	Office	DeskPro	375.27
P00302	Whiteboard Markers (8-pack) Plus XL	Office	DeskPro	325.48
P00303	Whiteboard Markers (8-pack) Plus	Office	DeskPro	63.58
P00304	Wireless Mouse S	Office	InkFlow	291.38
P00305	Office Scissors	Office	WriteRight	13.71
P00306	A4 Copy Paper 500	Office	ErgoCraft	417.10

	sheets - Silver		-	
P00307	Ergonomic Chair Lite M	Office	PaperMint	255.20
P00308	Gel Pens (10- pack)	Office	PaperMint	345.16
P00309	Binder 5cm	Office	InkFlow	107.18
P00310	Laptop Stand XL	Office	DeskPro	374.78
P00311	Laptop Stand	Office	WriteRight	113.70
P00312	Gel Pens (10- pack)	Office	PaperMint	106.07
P00313	Ergonomic Chair XL	Office	PaperMint	157.04
P00314	Ergonomic Chair - Red	Office	WriteRight	383.04
P00315	Fountain Pen	Office	WriteRight	470.73
P00316	Fountain Pen	Office	InkFlow	283.19
P00317	Laptop Stand	Office	InkFlow	402.72
P00318	Binder 5cm Pro	Office	WriteRight	179.74
P00319	Sticky Notes (12 pads) Lite	Office	DeskPro	11.96
P00320	Desk Lamp LED - White	Office	ErgoCraft	254.81
P00321	Fountain Pen 2.0 - Blue	Office	StapleStar	299.24
P00322	Wireless Mouse	Office	InkFlow	377.43

P00323	Laptop Stand - Blue	Office	DeskPro	474.00
P00324	Whiteboard Markers (8-pack) L	Office	InkFlow	442.41
P00325	Laptop Stand	Office	WriteRight	289.77
P00326	Ergonomic Chair Plus	Office	StapleStar	15.40
P00327	Fountain Pen	Office	PaperMint	444.00
P00328	Laptop Stand Pro - Blue	Office	StapleStar	146.01
P00329	Whiteboard Markers (8-pack)	Office	PaperMint	55.79
P00330	Notebook A5 Dotted	Office	InkFlow	449.83
P00331	Gel Pens (10-pack) Max	Office	WriteRight	264.97
P00332	A4 Copy Paper 500 sheets	Office	WriteRight	387.81
P00333	Notebook A5 Dotted 2.0 - White	Office	PaperMint	482.02
P00334	Laptop Stand Max L	Office	InkFlow	223.74
P00335	Wireless Mouse M	Office	StapleStar	87.62
P00336	Notebook A5 Dotted	Office	InkFlow	293.54

P00337	Whiteboard Markers (8-pack) Lite	Office	WriteRight	180.91
P00338	A4 Copy Paper 500 sheets	Office	DeskPro	33.80
P00339	Binder 5cm L	Office	DeskPro	322.87
P00340	Whiteboard Markers (8-pack) Max	Office	WriteRight	140.72
P00341	Ergonomic Chair S	Office	ErgoCraft	436.59
P00342	Ergonomic Chair	Office	WriteRight	11.92
P00343	Ergonomic Chair	Office	InkFlow	140.77
P00344	Laptop Stand	Office	PaperMint	407.77
P00345	Binder 5cm Pro	Office	WriteRight	483.72
P00346	Desk Lamp LED Pro - Blue	Office	ErgoCraft	361.23
P00347	Laptop Stand	Office	WriteRight	301.63
P00348	Laptop Stand	Office	PaperMint	487.10
P00349	A4 Copy Paper 500 sheets	Office	WriteRight	159.02
P00350	Office Scissors Pro	Office	StapleStar	80.20

	ITEM DESCRIPTION	DEPARTMENT	MANUFACTURER	UNIT PRICE
P00351	Dotted - Green	Office	WriteRight	215.47
P00352	A4 Copy Paper 500 sheets - Blue	Office	ErgoCraft	407.61
P00353	Sticky Notes (12 pads) 2.0	Office	StapleStar	432.63
P00354	Notebook A5 Dotted Lite - Black	Office	DeskPro	303.51
P00355	Office Scissors - Black	Office	PaperMint	280.93
P00356	Sticky Notes (12 pads)	Office	WriteRight	111.77
P00357	Fountain Pen S - Red	Office	PaperMint	128.95
P00358	Fountain Pen - Red	Office	StapleStar	320.53
P00359	Desk Lamp LED	Office	DeskPro	353.47
P00360	Sticky Notes (12 pads) 2.0	Office	StapleStar	478.13
P00361	Sticky Notes (12 pads) - White	Office	PaperMint	363.32
P00362	Ergonomic Chair	Office	DeskPro	281.69
P00363	Notebook A5 Dotted Plus	Office	WriteRight	214.26
	Whiteboard			

P00364	Markers (8-pack) Max - Blue	Office	DeskPro	33.05
P00365	Laptop Stand - Blue	Office	PaperMint	149.04
P00366	Whiteboard Markers (8-pack)	Office	StapleStar	30.94
P00367	Laptop Stand Max	Office	DeskPro	278.59
P00368	Whiteboard Markers (8-pack)	Office	DeskPro	440.39
P00369	Desk Lamp LED Pro	Office	ErgoCraft	479.67
P00370	A4 Copy Paper 500 sheets	Office	DeskPro	461.91
P00371	Fountain Pen Lite	Office	ErgoCraft	212.90
P00372	Binder 5cm	Office	PaperMint	210.97
P00373	Gel Pens (10-pack) Lite	Office	WriteRight	10.02
P00374	Desk Lamp LED L - Red	Office	DeskPro	258.07
P00375	Notebook A5 Dotted 2.0	Office	StapleStar	369.43
P00376	Fountain Pen	Office	ErgoCraft	235.63
P00377	Sticky Notes (12 pads) Pro	Office	StapleStar	172.52

P00378	Wireless Mouse	Office	PaperMint	240.21
P00379	Ergonomic Chair - Graphite	Office	StapleStar	290.36
P00380	Gel Pens (10-pack) 2.0	Office	WriteRight	162.49
P00381	Binder 5cm	Office	InkFlow	23.21
P00382	Laptop Stand Lite	Office	DeskPro	7.53
P00383	Laptop Stand Plus XL	Office	InkFlow	425.24
P00384	Binder 5cm - Graphite	Office	ErgoCraft	424.25
P00385	Ergonomic Chair	Office	ErgoCraft	159.54
P00386	Gel Pens (10-pack) Max - Blue	Office	PaperMint	338.80
P00387	Wireless Mouse - Green	Office	InkFlow	376.69
P00388	Sticky Notes (12 pads)	Office	WriteRight	369.50
P00389	Office Scissors	Office	StapleStar	186.82
P00390	Wireless Mouse Plus XL	Office	StapleStar	189.27
P00391	Laptop Stand - White	Office	PaperMint	476.86

P00392	Wireless Mouse Plus	Office	DeskPro	364.48
P00393	Whiteboard Markers (8-pack)	Office	InkFlow	415.87
P00394	Gel Pens (10-pack) XL	Office	ErgoCraft	271.79
P00395	Notebook A5 Dotted	Office	WriteRight	174.57
P00396	Gel Pens (10-pack) - Silver	Office	StapleStar	262.67
P00397	Ergonomic Chair S	Office	InkFlow	301.66
P00398	Notebook A5 Dotted 2.0	Office	InkFlow	202.18
P00399	A4 Copy Paper 500 sheets M - Black	Office	DeskPro	290.11
P00400	Ergonomic Chair	Office	StapleStar	437.29
P00401	Toaster 2-slice	Home & Kitchen	CozyNest	439.29
P00402	Cutlery Set (24 pcs) Pro	Home & Kitchen	CasaNova	688.76
P00403	Water Filter Jug 2.5L - Blue	Home & Kitchen	CozyNest	454.51
P00404	Ceramic Mug 350ml	Home & Kitchen	Hearthstone	566.11
-----	Air Fryer 4L	Home &	-----	-----

P00405	Pro XL	Kitchen	KitchenCrafter	580.22
P00406	Pressure Cooker 6L	Home & Kitchen	KitchenCrafter	82.37
P00407	Pressure Cooker 6L	Home & Kitchen	BrightGlow	356.45
P00408	Vacuum Cleaner Lite S	Home & Kitchen	Hearthstone	185.73
P00409	Glass Storage Set (10 pcs) Lite L - Blue	Home & Kitchen	Hearthstone	498.48
P00410	Vacuum Cleaner	Home & Kitchen	KitchenCrafter	640.77
P00411	Air Fryer 4L - Red	Home & Kitchen	CozyNest	598.29
P00412	Glass Storage Set (10 pcs)	Home & Kitchen	KitchenCrafter	187.39
P00413	Electric Kettle 1.7L Lite	Home & Kitchen	KitchenCrafter	276.75
P00414	Pressure Cooker 6L	Home & Kitchen	CozyNest	594.08
P00415	Blender 1.5L Lite	Home & Kitchen	CozyNest	258.79
P00416	Cutlery Set (24 pcs) - Graphite	Home & Kitchen	Hearthstone	130.55
P00417	Toaster 2-slice	Home & Kitchen	AquaPure	27.04
P00418	Ceramic Mug 350ml	Home & Kitchen	BrightGlow	441.46

P00419	Air Fryer 4L Pro XL - Graphite	Home & Kitchen	AquaPure	313.05
P00420	Stainless Steel Pan 28cm	Home & Kitchen	BrightGlow	131.87
P00421	Cutlery Set (24 pcs) M - Blue	Home & Kitchen	AquaPure	328.59
P00422	Vacuum Cleaner Lite	Home & Kitchen	CasaNova	187.63
P00423	Cutlery Set (24 pcs)	Home & Kitchen	Hearthstone	402.08
P00424	Pressure Cooker 6L	Home & Kitchen	KitchenCrafter	540.23
P00425	Cutlery Set (24 pcs)	Home & Kitchen	Hearthstone	462.70
P00426	Water Filter Jug 2.5L Plus	Home & Kitchen	Hearthstone	173.86
P00427	Air Fryer 4L	Home & Kitchen	CasaNova	572.27
P00428	Pressure Cooker 6L	Home & Kitchen	Hearthstone	180.18
P00429	Air Fryer 4L	Home & Kitchen	CasaNova	580.17
P00430	Electric Kettle 1.7L - Silver	Home & Kitchen	CozyNest	520.63
P00431	Glass Storage Set (10 pcs)	Home & Kitchen	KitchenCrafter	275.47
P00432	Vacuum Cleaner Plus	Home & Kitchen	AquaPure	199.48

	Cleaner Plus	Kitchen		
P00433	Stainless Steel Pan 28cm	Home & Kitchen	AquaPure	166.74
P00434	Stainless Steel Pan 28cm Pro	Home & Kitchen	BrightGlow	422.13
P00435	Ceramic Mug 350ml	Home & Kitchen	CozyNest	279.60
P00436	Glass Storage Set (10 pcs)	Home & Kitchen	CozyNest	174.71
P00437	Glass Storage Set (10 pcs) Plus	Home & Kitchen	Hearthstone	654.48
P00438	Blender 1.5L - White	Home & Kitchen	CozyNest	535.01
P00439	Chef Knife 20cm 2.0 - Blue	Home & Kitchen	KitchenCrafter	50.75
P00440	Pressure Cooker 6L - Green	Home & Kitchen	CasaNova	445.15
P00441	Blender 1.5L L	Home & Kitchen	KitchenCrafter	572.07
P00442	Toaster 2-slice	Home & Kitchen	KitchenCrafter	205.93
P00443	Glass Storage Set (10 pcs)	Home & Kitchen	Hearthstone	308.16
P00444	Blender 1.5L Plus	Home & Kitchen	KitchenCrafter	356.08
P00445	Blender 1.5L Lite	Home & Kitchen	AquaPure	145.83

P00446	Toaster 2-slice Lite	Home & Kitchen	BrightGlow	612.13
P00447	Stainless Steel Pan 28cm L	Home & Kitchen	BrightGlow	297.10
P00448	Ceramic Mug 350ml	Home & Kitchen	BrightGlow	59.74
P00449	Blender 1.5L	Home & Kitchen	CozyNest	51.54
P00450	Toaster 2-slice Max L	Home & Kitchen	CozyNest	148.36
P00451	Blender 1.5L Max	Home & Kitchen	AquaPure	691.34
P00452	Cutlery Set (24 pcs) 2.0	Home & Kitchen	Hearthstone	143.94
P00453	Vacuum Cleaner S - Silver	Home & Kitchen	KitchenCrafter	620.50
P00454	Blender 1.5L	Home & Kitchen	BrightGlow	453.88
P00455	Cutlery Set (24 pcs)	Home & Kitchen	BrightGlow	365.80
P00456	Toaster 2-slice M	Home & Kitchen	Hearthstone	170.10
P00457	Pressure Cooker 6L Pro S	Home & Kitchen	CasaNova	318.14
P00458	Glass Storage Set (10 pcs) XL	Home & Kitchen	KitchenCrafter	100.37

P00459	Cutlery Set (24 pcs)	Home & Kitchen	CozyNest	228.94
P00460	Cutlery Set (24 pcs) Lite - Silver	Home & Kitchen	KitchenCrafter	386.37
P00461	Air Fryer 4L	Home & Kitchen	BrightGlow	27.29
P00462	Vacuum Cleaner	Home & Kitchen	KitchenCrafter	432.95
P00463	Electric Kettle 1.7L	Home & Kitchen	BrightGlow	120.85
P00464	Vacuum Cleaner Pro	Home & Kitchen	AquaPure	270.29
P00465	Toaster 2-slice - White	Home & Kitchen	BrightGlow	241.97
P00466	Electric Kettle 1.7L L	Home & Kitchen	KitchenCrafter	657.73
P00467	Glass Storage Set (10 pcs) XL - White	Home & Kitchen	CasaNova	541.02
P00468	Ceramic Mug 350ml	Home & Kitchen	CozyNest	613.25
P00469	Toaster 2-slice L	Home & Kitchen	KitchenCrafter	475.19
P00470	Air Fryer 4L - Black	Home & Kitchen	KitchenCrafter	552.19
P00471	Toaster 2-slice	Home & Kitchen	Hearthstone	626.54
P00472	Chef Knife 20cm Plus - White	Home & Kitchen	KitchenCrafter	601.92

P00473	Chef Knife 20cm	Home & Kitchen	BrightGlow	465.78
P00474	Chef Knife 20cm Plus	Home & Kitchen	BrightGlow	245.50
P00475	Stainless Steel Pan 28cm Pro - Black	Home & Kitchen	AquaPure	672.47
P00476	Air Fryer 4L Pro M	Home & Kitchen	KitchenCrafter	268.52
P00477	Chef Knife 20cm - Silver	Home & Kitchen	CozyNest	499.61
P00478	Water Filter Jug 2.5L Plus L	Home & Kitchen	CasaNova	387.18
P00479	Toaster 2-slice Lite	Home & Kitchen	KitchenCrafter	32.66
P00480	Stainless Steel Pan 28cm Lite	Home & Kitchen	Hearthstone	279.81
P00481	Stainless Steel Pan 28cm	Home & Kitchen	AquaPure	637.17
P00482	Ceramic Mug 350ml	Home & Kitchen	KitchenCrafter	300.95
P00483	Electric Kettle 1.7L	Home & Kitchen	Hearthstone	582.93
P00484	Water Filter Jug 2.5L M - Silver	Home & Kitchen	Hearthstone	81.39
P00485	Glass Storage Set (10 pcs) -	Home &	AquaPure	626.74

	Product	Category	Brand	Price
	White	Kitchen		
P00486	Air Fryer 4L - Black	Home & Kitchen	Hearthstone	327.12
P00487	Vacuum Cleaner	Home & Kitchen	AquaPure	549.99
P00488	Blender 1.5L - Green	Home & Kitchen	CasaNova	356.52
P00489	Cutlery Set (24 pcs)	Home & Kitchen	Hearthstone	320.36
P00490	Electric Kettle 1.7L S	Home & Kitchen	BrightGlow	432.77
P00491	Toaster 2-slice Plus	Home & Kitchen	Hearthstone	308.19
P00492	Air Fryer 4L Lite - White	Home & Kitchen	BrightGlow	388.20
P00493	Ceramic Mug 350ml Pro - Red	Home & Kitchen	AquaPure	398.87
P00494	Cutlery Set (24 pcs) Lite - White	Home & Kitchen	KitchenCrafter	15.22
P00495	Toaster 2-slice Plus	Home & Kitchen	Hearthstone	182.22
P00496	Electric Kettle 1.7L - Graphite	Home & Kitchen	KitchenCrafter	242.41
P00497	Vacuum Cleaner	Home & Kitchen	AquaPure	463.21
P00498	Cutlery Set (24 pcs)	Home & Kitchen	KitchenCrafter	415.46

P00499	Water Filter Jug 2.5L M	Home & Kitchen	BrightGlow	347.72
P00500	Air Fryer 4L - Red	Home & Kitchen	CasaNova	169.56
P00501	Art Set 150 pcs	Toys	RoboPal	112.23
P00502	STEM Robot Kit M	Toys	KiddyLab	95.34
P00503	Remote Control Car Plus L	Toys	RoboPal	12.18
P00504	Science Kit - Volcano XL	Toys	SunnyPlay	7.48
P00505	Wooden Train Set	Toys	RoboPal	139.17
P00506	Plush Bear 30cm - Silver	Toys	KiddyLab	20.13
P00507	Wooden Train Set	Toys	SunnyPlay	60.80
P00508	STEM Robot Kit L - Green	Toys	SunnyPlay	197.49
P00509	Remote Control Car XL - Green	Toys	SunnyPlay	30.92
P00510	Puzzle 1000 pieces Max M	Toys	PlayVerse	146.49
P00511	Kite	Toys	RoboPal	93.28
P00512	STEM Robot Kit Lite XL	Toys	KiddyLab	148.57
P00513	Board Game	-	-	-

P00513	Classic Pro - Blue	Toys	SunnyPlay	54.10
P00514	RC Drone Mini Max	Toys	RoboPal	122.92
P00515	Wooden Train Set 2.0 XL	Toys	HappyBlocks	126.63
P00516	Wooden Train Set Plus XL	Toys	RoboPal	78.37
P00517	Marble Run 2.0	Toys	RoboPal	6.85
P00518	Kite	Toys	WonderWorks	66.24
P00519	RC Drone Mini S	Toys	SunnyPlay	43.60
P00520	Puzzle 1000 pieces M	Toys	HappyBlocks	90.82
P00521	Marble Run	Toys	WonderWorks	72.79
P00522	Plush Bear 30cm	Toys	RoboPal	117.81
P00523	Plush Bear 30cm - Black	Toys	KiddyLab	72.53
P00524	Plush Bear 30cm	Toys	KiddyLab	34.71
P00525	Building Blocks Set Pro - Black	Toys	PlayVerse	26.22
P00526	Wooden Train Set L - Black	Toys	SunnyPlay	5.91
	Puzzle 1000			

P00527	pieces XL	Toys	RoboPal	81.21
P00528	Plush Bear 30cm	Toys	SunnyPlay	24.16
P00529	Puzzle 1000 pieces 2.0 - Blue	Toys	HappyBlocks	76.61
P00530	Building Blocks Set Max - Blue	Toys	WonderWorks	27.86
P00531	Remote Control Car	Toys	WonderWorks	98.26
P00532	Marble Run Plus	Toys	HappyBlocks	121.42
P00533	Building Blocks Set	Toys	HappyBlocks	194.48
P00534	Puzzle 1000 pieces Lite	Toys	PlayVerse	107.09
P00535	Remote Control Car Plus XL	Toys	PlayVerse	26.95
P00536	Wooden Train Set Plus	Toys	PlayVerse	158.75
P00537	STEM Robot Kit	Toys	KiddyLab	28.11
P00538	Board Game Classic	Toys	HappyBlocks	122.75
P00539	RC Drone Mini	Toys	HappyBlocks	171.98
P00540	Puzzle 1000 pieces XL	Toys	HappyBlocks	128.12

P00541	RC Drone Mini - Green	Toys	KiddyLab	144.57
P00542	Remote Control Car 2.0 M	Toys	PlayVerse	130.79
P00543	RC Drone Mini M - Blue	Toys	HappyBlocks	130.50
P00544	Building Blocks Set L	Toys	RoboPal	21.34
P00545	Art Set 150 pcs 2.0 - Graphite	Toys	WonderWorks	190.98
P00546	STEM Robot Kit	Toys	WonderWorks	133.64
P00547	Remote Control Car	Toys	PlayVerse	59.19
P00548	RC Drone Mini Max	Toys	WonderWorks	64.32
P00549	RC Drone Mini Plus	Toys	SunnyPlay	40.65
P00550	RC Drone Mini	Toys	PlayVerse	41.19
P00551	Remote Control Car 2.0 - Black	Toys	KiddyLab	138.25
P00552	Wooden Train Set 2.0 - Silver	Toys	WonderWorks	154.51
P00553	STEM Robot Kit XL - Green	Toys	WonderWorks	191.16
P00554	Wooden Train Set 2.0 - Blue	-	WonderWorks	191.16

P00554	Train Set S - Blue	Toys	KiddyLab	29.25
P00555	RC Drone Mini - Graphite	Toys	RoboPal	111.84
P00556	Marble Run - Black	Toys	RoboPal	53.34
P00557	Puzzle 1000 pieces Max S	Toys	WonderWorks	171.71
P00558	Marble Run - Silver	Toys	WonderWorks	107.87
P00559	Building Blocks Set	Toys	PlayVerse	188.45
P00560	Building Blocks Set Lite	Toys	KiddyLab	44.12
P00561	STEM Robot Kit	Toys	SunnyPlay	144.77
P00562	Building Blocks Set - White	Toys	HappyBlocks	183.87
P00563	RC Drone Mini	Toys	SunnyPlay	55.65
P00564	Marble Run Plus - Blue	Toys	WonderWorks	129.41
P00565	Remote Control Car S	Toys	PlayVerse	167.50
P00566	Science Kit - Volcano	Toys	SunnyPlay	57.09
P00567	Building Blocks Set Max	Toys	SunnyPlay	46.45

P00568	Board Game Classic 2.0 - Red	Toys	KiddyLab	147.94
P00569	Remote Control Car L - Blue	Toys	WonderWorks	149.72
P00570	Puzzle 1000 pieces Pro - Blue	Toys	SunnyPlay	7.70
P00571	Marble Run Pro L	Toys	PlayVerse	120.21
P00572	Puzzle 1000 pieces Pro L - White	Toys	RoboPal	36.35
P00573	Science Kit - Volcano - Green	Toys	PlayVerse	199.29
P00574	Building Blocks Set	Toys	WonderWorks	193.17
P00575	Plush Bear 30cm Plus	Toys	SunnyPlay	113.46
P00576	STEM Robot Kit	Toys	RoboPal	189.36
P00577	Remote Control Car 2.0	Toys	WonderWorks	56.62
P00578	Remote Control Car Pro	Toys	WonderWorks	30.73
P00579	Building Blocks Set	Toys	RoboPal	158.78

P00580	Science Kit - Volcano S	Toys	HappyBlocks	181.20
P00581	Remote Control Car Lite	Toys	HappyBlocks	89.08
P00582	Building Blocks Set XL	Toys	SunnyPlay	107.64
P00583	RC Drone Mini Lite - Silver	Toys	SunnyPlay	156.23
P00584	Marble Run	Toys	HappyBlocks	157.81
P00585	Art Set 150 pcs Max M	Toys	SunnyPlay	133.27
P00586	Plush Bear 30cm Pro	Toys	WonderWorks	77.10
P00587	Remote Control Car Pro	Toys	HappyBlocks	193.25
P00588	Puzzle 1000 pieces Max	Toys	SunnyPlay	119.26
P00589	Wooden Train Set Max L - Graphite	Toys	RoboPal	127.01
P00590	Board Game Classic Plus - Blue	Toys	HappyBlocks	180.36
P00591	Plush Bear 30cm	Toys	RoboPal	86.82
P00592	RC Drone Mini	Toys	PlayVerse	103.86
P00593	Science Kit -	Toys	WonderWorks	126.00

	Volcano			
P00594	Wooden Train Set M	Toys	SunnyPlay	37.60
P00595	Puzzle 1000 pieces - Graphite	Toys	HappyBlocks	131.27
P00596	Remote Control Car Max L	Toys	PlayVerse	89.00
P00597	Building Blocks Set Lite	Toys	KiddyLab	140.49
P00598	STEM Robot Kit	Toys	SunnyPlay	136.91
P00599	Art Set 150 pcs Max	Toys	PlayVerse	96.77
P00600	Building Blocks Set	Toys	RoboPal	127.01
P00601	Bike Helmet Pro S	Sports	TrailRunner	165.61
P00602	Compression Shirt	Sports	TrailRunner	262.37
P00603	Basketball	Sports	AeroFlex	250.93
P00604	Tennis Racket - Black	Sports	WaveRider	58.12
P00605	Compression Shirt M - Graphite	Sports	PeakMotion	313.58
P00606	Bike Helmet L	Sports	WaveRider	32.14

P00607	Basketball	Sports	CoreAthlete	265.51
P00608	Fitness Tracker Pro	Sports	FitFoundry	142.49
P00609	Cycling Gloves Pro	Sports	PeakMotion	113.62
P00610	Basketball Max M	Sports	AeroFlex	40.44
P00611	Fitness Tracker XL	Sports	FitFoundry	228.43
P00612	Fitness Tracker	Sports	TrailRunner	172.12
P00613	Compression Shirt	Sports	TrailRunner	43.62
P00614	Compression Shirt Max	Sports	WaveRider	94.52
P00615	Bike Helmet	Sports	AeroFlex	75.40
P00616	Compression Shirt - Silver	Sports	CoreAthlete	128.94
P00617	Water Bottle 1L Pro M	Sports	WaveRider	323.96
P00618	Water Bottle 1L - White	Sports	PeakMotion	207.03
P00619	Running Shoes	Sports	AeroFlex	157.35
P00620	Compression Shirt	Sports	CoreAthlete	157.59
P00621	Bike Helmet	Sports	CoreAthlete	217.62
P00622	Basketball Lite L	Sports	WaveRider	317.17

P00623	Bike Helmet Plus L	Sports	WaveRider	57.39
P00624	Compression Shirt M	Sports	CoreAthlete	58.18
P00625	Compression Shirt	Sports	WaveRider	220.29
P00626	Running Shoes - Green	Sports	PeakMotion	196.77
P00627	Running Shoes	Sports	WaveRider	322.16
P00628	Water Bottle 1L Pro	Sports	CoreAthlete	291.05
P00629	Bike Helmet	Sports	TrailRunner	24.81
P00630	Football Size 5 Lite L	Sports	PeakMotion	310.37
P00631	Jump Rope XL	Sports	FitFoundry	158.82
P00632	Cycling Gloves	Sports	FitFoundry	199.80
P00633	Fitness Tracker 2.0 XL	Sports	AeroFlex	96.40
P00634	Jump Rope S	Sports	TrailRunner	67.87
P00635	Jump Rope	Sports	TrailRunner	66.84
P00636	Tennis Racket Pro	Sports	AeroFlex	42.03
P00637	Cycling Gloves - Red	Sports	WaveRider	249.39

P00638	Fitness Tracker Max	Sports	TrailRunner	285.70
P00639	Yoga Mat 6mm Lite - Blue	Sports	CoreAthlete	56.32
P00640	Dumbbells Pair 2x5kg	Sports	FitFoundry	44.64
P00641	Basketball S	Sports	AeroFlex	76.66
P00642	Dumbbells Pair 2x5kg 2.0	Sports	FitFoundry	186.95
P00643	Fitness Tracker	Sports	FitFoundry	249.69
P00644	Tennis Racket Plus	Sports	TrailRunner	65.43
P00645	Water Bottle 1L Pro	Sports	PeakMotion	273.54
P00646	Compression Shirt 2.0 - Black	Sports	AeroFlex	118.02
P00647	Cycling Gloves S	Sports	FitFoundry	110.51
P00648	Tennis Racket S	Sports	PeakMotion	309.01
P00649	Running Shoes	Sports	TrailRunner	123.14
P00650	Water Bottle 1L	Sports	CoreAthlete	286.07
P00651	Tennis Racket Lite - Graphite	Sports	TrailRunner	157.59

P00652	Cycling Gloves L - Blue	Sports	PeakMotion	104.26
P00653	Running Shoes Pro - Green	Sports	CoreAthlete	287.26
P00654	Football Size 5 Plus M	Sports	TrailRunner	132.22
P00655	Cycling Gloves S	Sports	WaveRider	224.58
P00656	Compression Shirt 2.0 M - Graphite	Sports	PeakMotion	59.44
P00657	Yoga Mat 6mm Lite - Blue	Sports	AeroFlex	165.61
P00658	Cycling Gloves	Sports	FitFoundry	109.42
P00659	Basketball	Sports	FitFoundry	253.86
P00660	Basketball Pro	Sports	WaveRider	229.68
P00661	Yoga Mat 6mm Max	Sports	AeroFlex	111.71
P00662	Running Shoes 2.0 S	Sports	AeroFlex	37.64
P00663	Water Bottle 1L	Sports	WaveRider	207.53
P00664	Jump Rope	Sports	CoreAthlete	204.48
P00665	Compression Shirt Plus	Sports	PeakMotion	218.33

P00666	Dumbbells Pair 2x5kg	Sports	WaveRider	294.43
P00667	Running Shoes Plus	Sports	TrailRunner	299.60
P00668	Compression Shirt Pro	Sports	CoreAthlete	103.98
P00669	Dumbbells Pair 2x5kg	Sports	TrailRunner	261.38
P00670	Compression Shirt - Silver	Sports	PeakMotion	50.54
P00671	Running Shoes Max M	Sports	WaveRider	101.03
P00672	Dumbbells Pair 2x5kg Pro - Blue	Sports	PeakMotion	108.67
P00673	Fitness Tracker Pro	Sports	CoreAthlete	162.90
P00674	Football Size 5 Plus S	Sports	PeakMotion	31.27
P00675	Tennis Racket Pro	Sports	PeakMotion	164.09
P00676	Football Size 5 Pro	Sports	AeroFlex	135.67
P00677	Basketball Lite	Sports	TrailRunner	322.08
P00678	Fitness Tracker Lite - Graphite	Sports	WaveRider	162.28
P00679	Running Shoes Pro - Green	Sports	TrailRunner	63.83

P00680	Basketball - Blue	Sports	CoreAthlete	193.29
P00681	Water Bottle 1L Lite	Sports	TrailRunner	166.01
P00682	Dumbbells Pair 2x5kg L	Sports	FitFoundry	291.58
P00683	Cycling Gloves - Green	Sports	FitFoundry	236.30
P00684	Cycling Gloves Pro - Blue	Sports	FitFoundry	30.44
P00685	Fitness Tracker Lite	Sports	WaveRider	94.38
P00686	Dumbbells Pair 2x5kg Max M	Sports	PeakMotion	339.06
P00687	Tennis Racket Lite	Sports	WaveRider	267.81
P00688	Water Bottle 1L - Blue	Sports	FitFoundry	51.50
P00689	Running Shoes Max	Sports	FitFoundry	58.35
P00690	Water Bottle 1L 2.0	Sports	FitFoundry	50.41
P00691	Water Bottle 1L 2.0	Sports	AeroFlex	321.26
P00692	Water Bottle 1L Pro	Sports	PeakMotion	163.06
P00693	Running Shoes	Sports	TrailRunner	299.56

P00694	Jump Rope	Sports	TrailRunner	48.40
P00695	Cycling Gloves Pro	Sports	TrailRunner	312.09
P00696	Cycling Gloves 2.0	Sports	TrailRunner	322.33
P00697	Tennis Racket 2.0 - Black	Sports	TrailRunner	320.34
P00698	Tennis Racket Pro	Sports	FitFoundry	58.95
P00699	Tennis Racket	Sports	TrailRunner	285.07
P00700	Football Size 5 Pro - Graphite	Sports	FitFoundry	318.47
P00701	Moisturizing Cream 50ml L - Black	Beauty	OceanMist	41.90
P00702	Moisturizing Cream 50ml Max	Beauty	OceanMist	88.33
P00703	Face Serum 30ml Max	Beauty	SilkAura	31.69
P00704	Conditioner 300ml 2.0 S - Green	Beauty	VelvetSkin	49.79
P00705	Hand Cream 75ml Max	Beauty	OceanMist	29.83
P00706	Lip Balm S	Beauty	SilkAura	86.17
P00707	Shampoo	Beauty	PureGlow	109.21

Product	Description	Category	Supplier	Cost
P00708	Sunscreen SPF50 100ml Max	Beauty	AurumCare	27.27
P00709	Hand Cream 75ml Plus - Black	Beauty	AurumCare	102.09
P00710	Hand Cream 75ml L - Red	Beauty	PureGlow	19.37
P00711	Micellar Water 400ml Plus L	Beauty	LunaLeaf	75.31
P00712	Facial Cleanser 200ml	Beauty	SilkAura	44.36
P00713	Face Serum 30ml L	Beauty	VelvetSkin	69.96
P00714	Hand Cream 75ml Plus - Green	Beauty	AurumCare	76.01
P00715	Face Mask Sheet	Beauty	PureGlow	19.68
P00716	Body Lotion 250ml - Silver	Beauty	PureGlow	81.96
P00717	Hand Cream 75ml	Beauty	PureGlow	2.73
P00718	Face Serum 30ml	Beauty	OceanMist	127.41
P00719	Moisturizing Cream 50ml Plus - Graphite	Beauty	SilkAura	106.20

P00720	Shampoo 300ml - Blue	Beauty	AurumCare	131.84
P00721	Body Lotion 250ml Pro	Beauty	PureGlow	21.40
P00722	Facial Cleanser 200ml - Silver	Beauty	VelvetSkin	11.71
P00723	Lip Balm XL - Black	Beauty	SilkAura	12.65
P00724	Hand Cream 75ml	Beauty	VelvetSkin	2.91
P00725	Face Serum 30ml Plus - Green	Beauty	SilkAura	60.39
P00726	Lip Balm	Beauty	SilkAura	65.05
P00727	Shampoo 300ml XL - Blue	Beauty	LunaLeaf	92.35
P00728	Moisturizing Cream 50ml Lite	Beauty	VelvetSkin	68.49
P00729	Micellar Water 400ml Pro M - Green	Beauty	SilkAura	78.72
P00730	Body Lotion 250ml	Beauty	VelvetSkin	98.43
P00731	Shampoo 300ml Plus L	Beauty	PureGlow	107.01
P00732	Body Lotion 250ml Plus S	Beauty	OceanMist	64.81
	Moisturizing			

P00733	Cream 50ml Lite	Beauty	SilkAura	77.40
P00734	Hand Cream 75ml Plus XL - Red	Beauty	AurumCare	136.27
P00735	Conditioner 300ml	Beauty	LunaLeaf	65.34
P00736	Body Lotion 250ml M	Beauty	VelvetSkin	61.09
P00737	Hair Oil 100ml - Green	Beauty	SilkAura	12.16
P00738	Lip Balm S	Beauty	LunaLeaf	50.76
P00739	Lip Balm Plus XL	Beauty	PureGlow	5.57
P00740	Body Lotion 250ml Pro - Black	Beauty	AurumCare	126.14
P00741	Face Mask Sheet Max	Beauty	VelvetSkin	138.50
P00742	Body Lotion 250ml Pro	Beauty	PureGlow	93.93
P00743	Face Serum 30ml XL	Beauty	PureGlow	39.40
P00744	Conditioner 300ml	Beauty	AurumCare	130.17
P00745	Body Lotion 250ml Pro S - Graphite	Beauty	SilkAura	4.49
P00746	Lip Balm Max - Black	Beauty	SilkAura	38.77

P00747	Micellar Water 400ml Pro	Beauty	SilkAura	129.80
P00748	Conditioner 300ml 2.0	Beauty	OceanMist	47.01
P00749	Micellar Water 400ml Pro	Beauty	AurumCare	100.06
P00750	Face Mask Sheet	Beauty	OceanMist	57.80
P00751	Face Mask Sheet XL	Beauty	LunaLeaf	9.92
P00752	Moisturizing Cream 50ml Max	Beauty	VelvetSkin	31.44
P00753	Body Lotion 250ml	Beauty	LunaLeaf	25.17
P00754	Shampoo 300ml Pro	Beauty	AurumCare	92.78
P00755	Moisturizing Cream 50ml Lite	Beauty	VelvetSkin	134.06
P00756	Hand Cream 75ml	Beauty	SilkAura	56.52
P00757	Hair Oil 100ml Lite S	Beauty	OceanMist	120.61
P00758	Face Mask Sheet L	Beauty	SilkAura	131.34
P00759	Conditioner 300ml	Beauty	VelvetSkin	92.75

P00760	Hair Oil 100ml	Beauty	AurumCare	30.29
P00761	Face Serum 30ml Plus M	Beauty	SilkAura	29.36
P00762	Moisturizing Cream 50ml 2.0 - Graphite	Beauty	SilkAura	92.41
P00763	Body Lotion 250ml Pro	Beauty	OceanMist	111.37
P00764	Micellar Water 400ml Max XL	Beauty	VelvetSkin	29.66
P00765	Hair Oil 100ml 2.0 - White	Beauty	AurumCare	127.31
P00766	Conditioner 300ml Pro	Beauty	PureGlow	88.22
P00767	Moisturizing Cream 50ml Pro - Green	Beauty	LunaLeaf	48.64
P00768	Shampoo 300ml Max	Beauty	AurumCare	78.94
P00769	Facial Cleanser 200ml Plus - Red	Beauty	PureGlow	92.86
P00770	Conditioner 300ml Pro L	Beauty	VelvetSkin	105.44
P00771	Conditioner 300ml XL	Beauty	SilkAura	32.34
P00772	Lip Balm	Beauty	VelvetSkin	89.61
	Facial			

P00773	Cleanser 200ml Max	Beauty	OceanMist	49.54
P00774	Hair Oil 100ml Pro - Blue	Beauty	LunaLeaf	71.11
P00775	Face Mask Sheet	Beauty	LunaLeaf	22.30
P00776	Lip Balm Max	Beauty	VelvetSkin	30.83
P00777	Sunscreen SPF50 100ml 2.0 - Green	Beauty	SilkAura	135.21
P00778	Moisturizing Cream 50ml - Red	Beauty	AurumCare	95.86
P00779	Facial Cleanser 200ml	Beauty	PureGlow	31.85
P00780	Moisturizing Cream 50ml Pro XL	Beauty	AurumCare	72.28
P00781	Body Lotion 250ml	Beauty	SilkAura	16.46
P00782	Facial Cleanser 200ml	Beauty	LunaLeaf	19.85
P00783	Sunscreen SPF50 100ml 2.0 L	Beauty	SilkAura	57.04
P00784	Moisturizing Cream 50ml	Beauty	AurumCare	120.47
P00785	Conditioner 300ml L	Beauty	VelvetSkin	38.51

P00786	Body Lotion 250ml - Black	Beauty	LunaLeaf	79.95
P00787	Sunscreen SPF50 100ml XL	Beauty	PureGlow	89.45
P00788	Lip Balm	Beauty	LunaLeaf	24.46
P00789	Hair Oil 100ml M - Silver	Beauty	LunaLeaf	62.47
P00790	Face Mask Sheet	Beauty	OceanMist	16.06
P00791	Body Lotion 250ml - Silver	Beauty	AurumCare	81.83
P00792	Sunscreen SPF50 100ml	Beauty	PureGlow	39.84
P00793	Facial Cleanser 200ml S	Beauty	LunaLeaf	46.03
P00794	Hand Cream 75ml 2.0	Beauty	OceanMist	82.19
P00795	Moisturizing Cream 50ml	Beauty	LunaLeaf	124.66
P00796	Facial Cleanser 200ml Plus	Beauty	VelvetSkin	134.25
P00797	Sunscreen SPF50 100ml	Beauty	VelvetSkin	114.51
P00798	Hand Cream 75ml Plus	Beauty	VelvetSkin	139.58
P00799	Conditioner	Beauty	VelvetSkin	96.41

	300ml Max S			
P00800	Face Mask Sheet	Beauty	LunaLeaf	135.28
P00801	Cookbook - Quick Meals - Red	Books	Atlas Editions	63.37
P00802	Paperback Novel Lite M	Books	Orion Leaf	16.39
P00803	Cookbook - Quick Meals Plus M - Blue	Books	Orion Leaf	71.91
P00804	Poetry Collection Pro - White	Books	Silver Quill	27.97
P00805	Mystery Thriller	Books	Silver Quill	67.46
P00806	Poetry Collection Pro	Books	Northwind Press	42.52
P00807	Travel Guide: Japan S	Books	Atlas Editions	78.86
P00808	Poetry Collection - Green	Books	Atlas Editions	29.38
P00809	Beginners German Max	Books	Atlas Editions	51.71
P00810	Business Strategy 101 M	Books	OpenPage	50.87
P00811	Poetry Collection XL	Books	Northwind Press	72.06

P00812	Cookbook - Quick Meals Plus	Books	Sunset House	15.60
P00813	Mystery Thriller Pro M - Black	Books	Northwind Press	66.42
P00814	Short Stories Lite	Books	Sunset House	54.74
P00815	Short Stories Plus	Books	Silver Quill	44.93
P00816	Short Stories Max S - Blue	Books	Northwind Press	9.48
P00817	Travel Guide: Japan Max	Books	Sunset House	17.83
P00818	History of Art M - Green	Books	OpenPage	37.96
P00819	Business Strategy 101 - Red	Books	Northwind Press	9.58
P00820	Beginners German Lite	Books	OpenPage	48.86
P00821	Beginners German	Books	Northwind Press	73.32
P00822	Data Science Handbook	Books	Atlas Editions	53.68
P00823	History of Art Pro	Books	Northwind Press	13.64
P00824	Cookbook - Quick Meals XL	Books	Northwind Press	30.98
P00825	Beginners German M	Books	Atlas Editions	71.44

P00826	Short Stories L - Graphite	Books	OpenPage	13.13
P00827	Beginners German Max	Books	Atlas Editions	63.57
P00828	Paperback Novel Pro S	Books	OpenPage	44.73
P00829	Poetry Collection M	Books	Atlas Editions	19.62
P00830	Short Stories - Graphite	Books	Orion Leaf	17.45
P00831	Beginners German Lite	Books	Silver Quill	29.29
P00832	Travel Guide: Japan Pro - Graphite	Books	Silver Quill	6.19
P00833	Cookbook - Quick Meals Lite	Books	OpenPage	36.86
P00834	Mystery Thriller M	Books	OpenPage	62.81
P00835	Data Science Handbook	Books	Northwind Press	56.05
P00836	Data Science Handbook	Books	Northwind Press	37.35
P00837	Business Strategy 101	Books	Silver Quill	78.31
P00838	Paperback Novel	Books	Silver Quill	65.37
P00839	Poetry Collection	Books	Silver Quill	68.19

P00840	Travel Guide: Japan	Books	OpenPage	16.19
P00841	Poetry Collection - Silver	Books	Silver Quill	7.06
P00842	Cookbook - Quick Meals Max - Blue	Books	Sunset House	55.64
P00843	Travel Guide: Japan Plus - Black	Books	Atlas Editions	42.16
P00844	Business Strategy 101 - Graphite	Books	Silver Quill	54.11
P00845	Paperback Novel M	Books	Orion Leaf	57.08
P00846	Short Stories Max M	Books	Atlas Editions	33.41
P00847	Travel Guide: Japan 2.0	Books	Orion Leaf	39.20
P00848	Business Strategy 101 Pro M	Books	Atlas Editions	25.58
P00849	Data Science Handbook	Books	Sunset House	22.21
P00850	Poetry Collection	Books	Orion Leaf	70.29
P00851	Mystery Thriller Pro L	Books	Silver Quill	72.78
P00852	Data Science Handbook	Books	Orion Leaf	49.92

P00853	Business Strategy 101 XL - Blue	Books	Atlas Editions	26.11
P00854	Poetry Collection	Books	Silver Quill	70.25
P00855	Travel Guide: Japan M - White	Books	Sunset House	12.73
P00856	Business Strategy 101	Books	Atlas Editions	77.06
P00857	Business Strategy 101 - Green	Books	Sunset House	27.66
P00858	Poetry Collection XL	Books	Atlas Editions	30.12
P00859	Data Science Handbook	Books	OpenPage	47.29
P00860	Business Strategy 101 - Silver	Books	Silver Quill	22.31
P00861	History of Art XL	Books	OpenPage	22.03
P00862	Fantasy Epic Vol. 1 M - Silver	Books	Orion Leaf	59.63
P00863	Data Science Handbook Pro	Books	Atlas Editions	7.80
P00864	Data Science Handbook - Green	Books	Atlas Editions	51.32
	Paperback	.		

P00865	Novel Pro S	Books	OpenPage	48.38
P00866	Short Stories - Black	Books	Orion Leaf	11.82
P00867	History of Art	Books	Silver Quill	31.45
P00868	Data Science Handbook L	Books	Silver Quill	29.71
P00869	Childrens Stories M	Books	Orion Leaf	15.60
P00870	Poetry Collection - Green	Books	Silver Quill	26.69
P00871	Mystery Thriller - Silver	Books	Orion Leaf	38.54
P00872	Paperback Novel	Books	Sunset House	23.78
P00873	Beginners German	Books	Northwind Press	46.69
P00874	Paperback Novel	Books	OpenPage	64.93
P00875	Mystery Thriller Lite	Books	Silver Quill	47.53
P00876	Short Stories	Books	Orion Leaf	21.19
P00877	Cookbook - Quick Meals Lite - Silver	Books	Atlas Editions	69.33
P00878	Cookbook - Quick Meals 2.0	Books	Northwind Press	53.36
	Fantasy Epic			

P00879	Vol. 1 - Graphite	Books	Atlas Editions	52.10
P00880	History of Art - Silver	Books	OpenPage	28.01
P00881	Childrens Stories 2.0 S	Books	Northwind Press	11.01
P00882	Fantasy Epic Vol. 1	Books	Silver Quill	27.10
P00883	History of Art	Books	Sunset House	36.85
P00884	Data Science Handbook S	Books	Orion Leaf	28.54
P00885	Childrens Stories Plus	Books	Atlas Editions	43.09
P00886	Poetry Collection Lite	Books	OpenPage	77.43
P00887	History of Art	Books	OpenPage	66.04
P00888	Poetry Collection - Graphite	Books	Orion Leaf	24.78
P00889	Mystery Thriller 2.0 XL	Books	Silver Quill	29.64
P00890	Data Science Handbook	Books	Orion Leaf	75.33
P00891	Poetry Collection Lite M	Books	Sunset House	9.56
P00892	Travel Guide: Japan Max	Books	Atlas Editions	11.59
P00893	Travel Guide: Japan Max	Books	Atlas Editions	19.70

	Japan			
P00894	Poetry Collection Max S	Books	Orion Leaf	28.49
P00895	Data Science Handbook	Books	Atlas Editions	28.22
P00896	Beginners German XL	Books	Northwind Press	22.43
P00897	Paperback Novel - Blue	Books	Northwind Press	6.54
P00898	Travel Guide: Japan XL	Books	Northwind Press	48.87
P00899	Childrens Stories Pro	Books	Atlas Editions	70.79
P00900	Data Science Handbook	Books	Atlas Editions	76.67
P00901	Aquarium Filter S - White	Pets	WildFeast	104.36
P00902	Bird Seed 1kg Max	Pets	WildFeast	93.18
P00903	Pet Bed Medium L	Pets	Paw&Co	120.70
P00904	Small Animal Hay 1kg	Pets	PetNest	60.02
P00905	Pet Shampoo 250ml	Pets	PetNest	180.44
P00906	Dry Dog Food 5kg	Pets	Paw&Co	92.58
P00907	Pet Shampoo 250ml M	Pets	FurryFriends	141.09

P00908	Dog Leash Nylon	Pets	PetNest	172.76
P00909	Pet Treats Mixed 500g M - White	Pets	AquaPets	147.83
P00910	Pet Treats Mixed 500g Lite XL - Red	Pets	HappyTail	19.70
P00911	Aquarium Filter L - White	Pets	Paw&Co	98.30
P00912	Pet Treats Mixed 500g	Pets	HappyTail	81.86
P00913	Bird Seed 1kg 2.0 - Blue	Pets	PetNest	23.38
P00914	Bird Seed 1kg 2.0	Pets	WildFeast	59.12
P00915	Pet Shampoo 250ml Plus - Blue	Pets	FurryFriends	102.17
P00916	Dog Leash Nylon Max - Red	Pets	HappyTail	75.98
P00917	Aquarium Filter	Pets	FurryFriends	60.21
P00918	Pet Shampoo 250ml 2.0 - Silver	Pets	FurryFriends	141.82
P00919	Aquarium Filter	Pets	PetNest	67.23
P00920	Pet Shampoo	Pets	Paw&Co	97.27

P00920	250ml Plus	Pets	Paw&Co	31.21
P00921	Cat Litter 10L - Graphite	Pets	PetNest	116.62
P00922	Pet Treats Mixed 500g Max	Pets	WildFeast	97.41
P00923	Cat Scratching Post Pro	Pets	PetNest	100.86
P00924	Cat Litter 10L 2.0 S	Pets	Paw&Co	160.83
P00925	Aquarium Filter Pro M - Blue	Pets	AquaPets	57.42
P00926	Pet Treats Mixed 500g Pro	Pets	AquaPets	13.42
P00927	Cat Litter 10L	Pets	WildFeast	113.46
P00928	Dog Leash Nylon Plus - Green	Pets	HappyTail	21.56
P00929	Bird Seed 1kg Lite	Pets	AquaPets	57.36
P00930	Pet Bed Medium Plus XL	Pets	AquaPets	55.81
P00931	Aquarium Filter Pro	Pets	AquaPets	172.05
P00932	Cat Litter 10L L	Pets	FurryFriends	76.39
P00933	Aquarium Filter	Pets	Paw&Co	48.62

	Filter			
P00934	Bird Seed 1kg	Pets	FurryFriends	13.52
P00935	Cat Litter 10L M - Green	Pets	PetNest	194.53
P00936	Pet Bed Medium	Pets	FurryFriends	136.77
P00937	Dog Leash Nylon	Pets	PetNest	140.62
P00938	Aquarium Filter	Pets	Paw&Co	29.24
P00939	Bird Seed 1kg Plus	Pets	AquaPets	132.10
P00940	Pet Shampoo 250ml	Pets	WildFeast	178.36
P00941	Pet Treats Mixed 500g M	Pets	FurryFriends	10.53
P00942	Dog Leash Nylon L	Pets	PetNest	59.81
P00943	Aquarium Filter Max M - Green	Pets	HappyTail	50.26
P00944	Fish Flakes 200g S - Blue	Pets	PetNest	184.90
P00945	Bird Seed 1kg	Pets	WildFeast	199.08
P00946	Pet Shampoo 250ml Pro - Red	Pets	WildFeast	26.75
P00947	Cat Scratching Post Plus	Pets	Paw&Co	147.88

P00948	Cat Litter 10L M	Pets	PetNest	121.48
P00949	Pet Shampoo 250ml XL - Black	Pets	AquaPets	38.94
P00950	Cat Litter 10L	Pets	Paw&Co	62.16
P00951	Pet Shampoo 250ml - Red	Pets	HappyTail	173.80
P00952	Pet Bed Medium Max XL	Pets	PetNest	42.11
P00953	Small Animal Hay 1kg 2.0	Pets	HappyTail	61.52
P00954	Pet Bed Medium Plus	Pets	WildFeast	3.95
P00955	Dry Dog Food 5kg	Pets	Paw&Co	111.94
P00956	Bird Seed 1kg Pro S - Silver	Pets	Paw&Co	4.98
P00957	Pet Treats Mixed 500g	Pets	Paw&Co	183.42
P00958	Pet Bed Medium L	Pets	HappyTail	108.12
P00959	Cat Toys (6-pack)	Pets	HappyTail	189.39
P00960	Pet Bed Medium	Pets	AquaPets	18.26
P00961	Bird Seed 1kg S - Silver	Pets	WildFeast	142.56
P00962	Dog Leash	Pets	WildFeast	82.43

Product ID	Product Name	Category	Supplier	Unit Price
P00963	Pet Shampoo 250ml Max - Green	Pets	WildFeast	177.81
P00964	Aquarium Filter	Pets	PetNest	193.86
P00965	Pet Bed Medium	Pets	PetNest	104.27
P00966	Cat Litter 10L Plus	Pets	HappyTail	52.97
P00967	Cat Litter 10L 2.0 XL	Pets	AquaPets	185.24
P00968	Dog Leash Nylon - Silver	Pets	WildFeast	17.14
P00969	Aquarium Filter XL	Pets	WildFeast	47.83
P00970	Cat Scratching Post - Red	Pets	FurryFriends	162.26
P00971	Pet Bed Medium Pro - Red	Pets	WildFeast	144.36
P00972	Cat Toys (6- pack) 2.0 XL - Red	Pets	PetNest	97.78
P00973	Fish Flakes 200g M	Pets	PetNest	120.56
P00974	Dog Leash Nylon Lite	Pets	WildFeast	104.69
P00975	Aquarium Filter - Blue	Pets	HappyTail	5.79

P00976	Pet Shampoo 250ml M	Pets	FurryFriends	45.09
P00977	Pet Bed Medium - Silver	Pets	PetNest	193.48
P00978	Dry Dog Food 5kg 2.0	Pets	HappyTail	80.88
P00979	Fish Flakes 200g	Pets	HappyTail	68.59
P00980	Small Animal Hay 1kg	Pets	AquaPets	111.22
P00981	Bird Seed 1kg 2.0 - Blue	Pets	PetNest	124.98
P00982	Cat Litter 10L Lite S	Pets	Paw&Co	64.17
P00983	Bird Seed 1kg - White	Pets	WildFeast	113.43
P00984	Small Animal Hay 1kg Plus	Pets	Paw&Co	136.54
P00985	Pet Treats Mixed 500g Pro S - Black	Pets	AquaPets	181.96
P00986	Fish Flakes 200g Max XL	Pets	PetNest	31.03
P00987	Pet Treats Mixed 500g	Pets	AquaPets	143.78
P00988	Pet Treats Mixed 500g	Pets	HappyTail	99.18
P00989	Small Animal Hay 1kg L - Silver	Pets	HappyTail	32.19

P00990	Cat Litter 10L	Pets	Paw&Co	196.00
P00991	Dry Dog Food 5kg Max XL	Pets	WildFeast	55.45
P00992	Cat Scratching Post Pro	Pets	FurryFriends	99.74
P00993	Cat Scratching Post Plus	Pets	FurryFriends	5.83
P00994	Aquarium Filter L	Pets	WildFeast	189.23
P00995	Bird Seed 1kg	Pets	HappyTail	4.30
P00996	Pet Bed Medium Lite S	Pets	PetNest	61.77
P00997	Pet Shampoo 250ml Plus - White	Pets	AquaPets	111.93
P00998	Dog Leash Nylon XL	Pets	HappyTail	148.29
P00999	Dog Leash Nylon 2.0 - Silver	Pets	FurryFriends	193.92
P01000	Bird Seed 1kg	Pets	Paw&Co	170.70

Eine typische Produkt-CSV mit ID, Name, Kategorie, Preis, Lagerbestand. Nichts Besonderes – aber schauen wir uns die Performance an!

#### Naive Implementierung: Lineare Suche

```
1 const response = await fetch("https://raw.githubusercontent.com/andre-dietrich/Datenbankensysteme-Vorlesung/refs/heads/main/assets/data/products.csv");
```

```

2 const text = await response.text();
3 const rows = Papa.parse(text, { header: true }).data;
4
5 function searchProductById(productId) {
6   const start = performance.now();
7   let found = null;
8
9   // Zeilenweise durchsuchen (O(n)!)
10  for (let i = 0; i < rows.length; i++) {
11    sleep(1); // Simuliere Latenz (I/O, Parsing)
12    if (rows[i].product_id == productId) {
13      found = rows[i];
14      break;
15    }
16  }
17
18  const end = performance.now();
19
20  if (!found) {
21    console.log("✗ Produkt nicht gefunden");
22  } else {
23    console.log("✓ Produkt gefunden:", JSON.stringify(found, null, 2));
24  }
25
26  console.log("⌚ Zeit:", (end - start).toFixed(2), "ms");
27}
28
29 searchProductById("P00567");

```

✓ Produkt gefunden: {  
 "product\_id": "P00567",  
 "name": "Building Blocks Set Max",  
 "category": "Toys",  
 "brand": "SunnyPlay",  
 "price": "46.45",  
 "stock": "257",  
 "rating": "4.7",  
 "created\_at": "2025-03-06"  
 }  
⌚ Zeit: 566.50 ms

Probiert verschiedene Produkt-IDs aus! Beobachtet: Je weiter hinten das Produkt in der Datei steht, desto länger dauert die Suche. Im schlimmsten Fall müssen wir die gesamte Datei durchsuchen!

**Problem:** Lineare Suche =  $O(n)$  Zeitkomplexität

- 1.000 Produkte? Noch ok (~1 Sekunde)
- 1.000.000 Produkte? **Nicht mehr praktikabel (~17 Minuten)**
- **Skalierungsproblem:** Verdoppelung der Daten = Verdoppelung der Suchzeit

## Big-O Notation: Kurzer Exkurs

Big-O Notation ist eine mathematische Notation, die verwendet wird, um die Leistung oder Komplexität eines Algorithmus zu beschreiben. Sie konzentriert sich auf das Worst-Case-Szenario und ignoriert Konstanten und niedrigere Ordnungstermine. Aber was bedeutet das konkret?

### Alltags-Analogie: Bücher im Regal finden

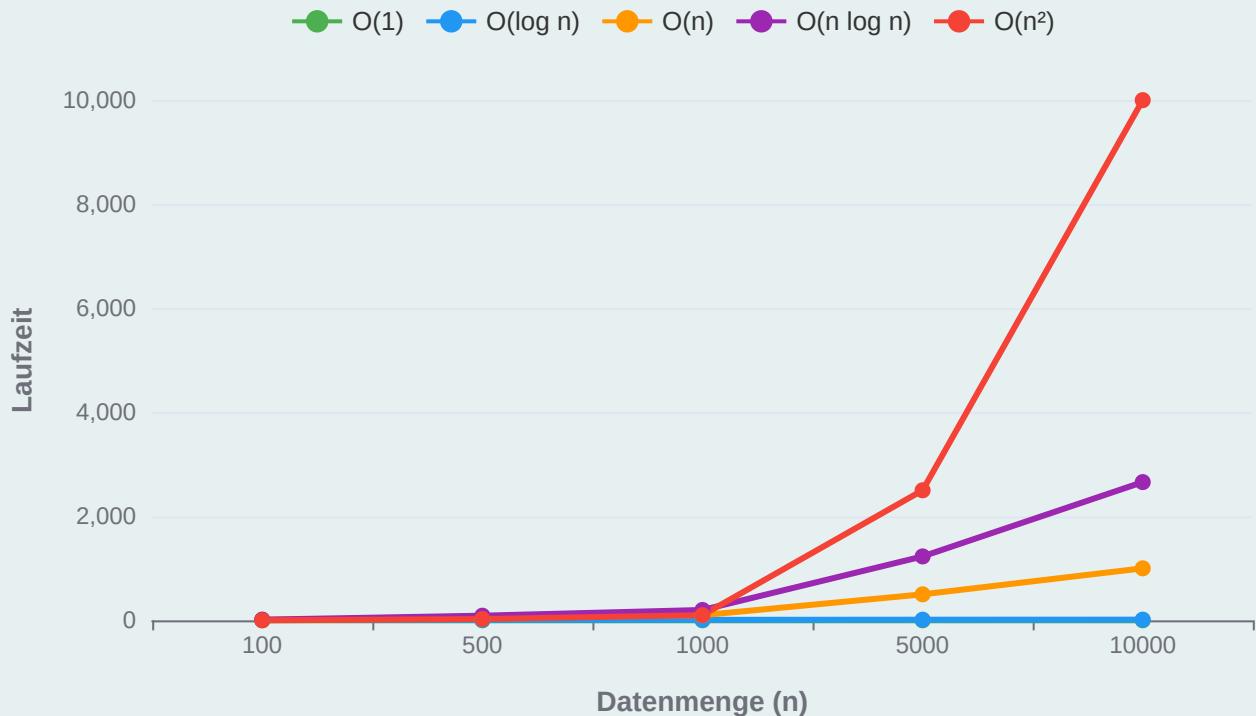
Stellt euch vor, ihr sucht ein bestimmtes Buch in einem Regal:

- **$O(1)$  – Konstant:** Ihr wisst genau, wo das Buch steht (Position 42). Egal ob 10 oder 10.000 Bücher im Regal – ihr greift direkt zu. **Wie ein Hash-Lookup!**
- **$O(\log n)$  – Logarithmisch:** Die Bücher sind alphabetisch sortiert. Ihr schaut in die Mitte, wisst ob links oder rechts, halbiert wieder – wie beim Telefonbuch. Bei 1.000 Büchern nur ~10 Schritte!
- **$O(n)$  – Linear:** Bücher sind unsortiert. Ihr müsst jedes einzelne Buch durchgehen. 1.000 Bücher = 1.000 Checks im Worst Case. **Wie CSV-Scan!**
- **$O(n \log n)$  – Linearithmisch:** Ihr wollt alle Bücher alphabetisch sortieren (z.B. mit MergeSort). Jedes Buch muss mindestens einmal angefasst werden ( $n$ )( $n$ ), und dabei werden  $\log n$  Vergleichsebenen durchlaufen. Langsamer als  $O(n)O(n)$ , aber viel schneller als  $O(n^2)O(n^2)$ !
- **$O(n^2)$  – Quadratisch:** Ihr vergleicht jedes Buch mit jedem anderen (z.B. Duplikate finden). Bei 1.000 Büchern = 1.000.000 Vergleiche!

Diese Analogie macht klar: Die Wahl der richtigen Datenstruktur und des richtigen Algorithmus kann den Unterschied zwischen Millisekunden und Stunden ausmachen!

### Visualisierung: Wachstum bei steigender Datenmenge

## Big-O Komplexität: Laufzeit vs. Datenmenge



Seht ihr den Unterschied?  $O(1)$  bleibt flach – egal wie viele Daten.  $O(n^2)$  explodiert förmlich! Das ist der Grund, warum Hash-Tables so wertvoll sind. Bewegt die Maus über die Linien, um die genauen Werte zu sehen!

Die häufigsten Big-O-Klassen im Überblick

Komplexität	Name	Beispiel	100 Elemente	1.000 Elemente
$O(1)O(1)$	Konstant	Hash-Lookup, Array[index]	1 ms	1 m
$O(\log n)$ $O(\log n)$	Logarithmisch	Binäre Suche, Baum-Traversierung	~7 ms	~10 m
$O(n)O(n)$	Linear	Array durchsuchen, CSV scannen	100 ms	1
$O(n \log n)$ $O(n \log n)$	Linearithmisch	Gute Sortieralgorithmen (Mergesort)	~700 ms	~10
$O(n^2)O(n^2)$	Quadratisch	Nested Loops, Bubble Sort	10 s	~17 mi

Diese Tabelle zeigt: Ab einer bestimmten Datenmenge wird der Algorithmus zum Flaschenhals.  $O(n^2)O(n^2)$  ist bei 1 Million Elementen praktisch unbenutzbar!

### Konkrete Code-Beispiele

```
// O(1) - Konstant: Direkter Zugriff
const product = db["P00567"]; // Immer gleich schnell!

// O(n) - Linear: Alle Elemente durchgehen
for (let i = 0; i < products.length; i++) {
  if (products[i].id === "P00567") return products[i];
}

// O(log n) - Logarithmisch: Binäre Suche (nur bei sortierten Daten!)
function binarySearch(sortedArray, target) {
  let left = 0, right = sortedArray.length - 1;
  while (left <= right) {
    const mid = Math.floor((left + right) / 2);
    if (sortedArray[mid] === target) return mid;
    if (sortedArray[mid] < target) left = mid + 1;
    else right = mid - 1;
  }
  return -1;
}
```

```
// O(n2) - Quadratisch: Verschachtelte Loops
for (let i = 0; i < products.length; i++) {
  for (let j = 0; j < products.length; j++) {
    // Vergleiche jedes Produkt mit jedem anderen
  }
}
```

Der Code macht den Unterschied deutlich: Ein Hash-Lookup ist eine einzige Zeile. Eine verschachtelte Schleife muss potentiell Millionen von Operationen durchführen!

**Merkregel:** Je flacher die Kurve, desto besser skaliert der Algorithmus!

- **Gut:**  $O(1)O(1)$ ,  $O(\log n)O(\log n)$
- **Akzeptabel:**  $O(n)O(n)$ ,  $O(n \log n)O(n \log n)$
- **Problematisch:**  $O(n^2)O(n^2)$ ,  $O(2^n)O(2^n)$ ,  $O(n!)O(n!)$

Big-O Cheat Sheet: <https://www.bigocheatsheet.com/>

## Die Lösung: Hash-basierter Index (Key-Value!)

Jetzt kommt der Aha-Moment: Was wäre, wenn wir die CSV einmal einlesen, aber dann einen Index bauen? Genau das macht ein Key-Value Store!

Optimierte Implementierung: Hash-Lookup

```
1 const response = await fetch("https://raw.githubusercontent.com/andre-dietrich/Datenbankensysteme-Vorlesung/refs/heads/main/assets/data/products.csv");
2 const text = await response.text();
3 const rows = Papa.parse(text, { header: true }).data;
4
5 // Index bauen: Key = product_id, Value = komplettes Produkt-Objekt
6 const db = {};
7 rows.forEach((row) => {
8   db[row.product_id] = row;
9 });
10
11 function searchProductById(productId) {
12   const start = performance.now();
13
14   // O(1) Lookup! 🚀
15   sleep(1); // Simuliere minimale Latenz
16   const product = db[productId];
17
18   const end = performance.now();
19
20   if (product === undefined) {
21     return null;
22   }
23
24   const duration = end - start;
25
26   console.log(`Search time: ${duration}ms`);
27
28   return product;
29 }
```

```

20  }   if (product === undefined) {
21      console.log("✅ Produkt gefunden:", JSON.stringify(product, null,
22  } else {
23      console.log("❌ Produkt nicht gefunden");
24  }
25
26  console.log("⌚ Zeit:", (end - start).toFixed(2), "ms");
27 }
28
29 searchProductById("P00567");

```

Identifier 'response' has already been declared

Boom! Von mehreren Sekunden auf unter 1 Millisekunde! Der Index macht den Unterschied: Wir zahlen einmal den Preis fürs Einlesen und Indexieren, danach sind alle Lookups blitzschnell.

### Performance-Vergleich

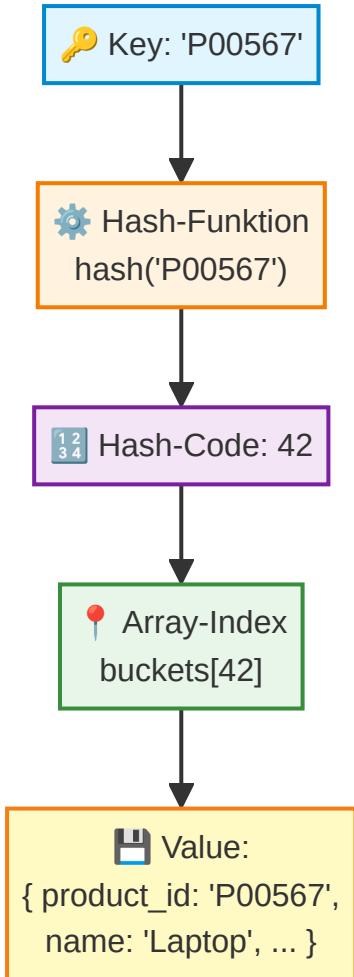
Methode	Zeitkomplexität	1.000 Produkte	1.000.000 Produkte
Lineare Suche	$O(n)$	1s 1s	17min 17min
Hash-Lookup	$O(1)$	< 1ms < 1ms	< 1ms < 1ms

Das ist die Superkraft von Key-Value: Konstante Zugriffszeit, egal wie viele Daten ihr habt!

## Wie funktioniert das intern?

Aber wie kann ein Hash-basierter Index so schnell sein? Die Antwort: Hash-Funktionen! Schauen wir uns an, wie das im Detail funktioniert.

### Das Prinzip: Von Key zu Array-Index



Eine Hash-Funktion nimmt einen beliebigen Key (String, Number) und berechnet daraus eine Zahl – den Hash-Code. Diese Zahl wird als Array-Index verwendet. Direkter Array-Zugriff ist  $O(1)$ !

### Schritt-für-Schritt: Einfache Hash-Table Implementation

```

1 // Vereinfachte Hash-Table (zum Verständnis)
2 class SimpleHashTable {
3     constructor(size = 100) {
4         this.buckets = new Array(size); // Array mit 100 Slots
5         this.size = size;
6     }
7
8     // Hash-Funktion: String → Number
9     hash(key) {
10        let hashValue = 0;
11        for (let i = 0; i < key.length; i++) {
12            hashValue += key.charCodeAt(i); // ASCII-Wert summieren
13        }
14        return hashValue % this.size; // Modulo für Array-Größe
15    }
16
17    // SET: Key-Value speichern
18    set(key, value) {

```

```

19  const index = this.hash(key);
20  console.log(`SET "${key}" → Hash: ${index}`);
21  this.buckets[index] = { key, value }; // Vereinfacht!
22 }
23
24 // GET: Value abrufen
25 get(key) {
26   const index = this.hash(key);
27   console.log(`GET "${key}" → Hash: ${index}`);
28   const bucket = this.buckets[index];
29   return bucket ? bucket.value : undefined;
30 }
31 }
32
33 // Demo
34 const hashTable = new SimpleHashTable(10);
35
36 hashTable.set("P00567", { name: "Laptop", price: 999 });
37 hashTable.set("P00123", { name: "Mouse", price: 25 });
38 hashTable.set("P00999", { name: "Keyboard", price: 75 });
39
40 console.log("\n--- Lookups ---");
41 const laptop = hashTable.get("P00567");
42 console.log("Found:", laptop);
43
44 const mouse = hashTable.get("P00123");
45 console.log("Found:", mouse);
46
47 // Nicht vorhanden
48 const unknown = hashTable.get("P99999");
49 console.log("Found:", unknown);

```

```

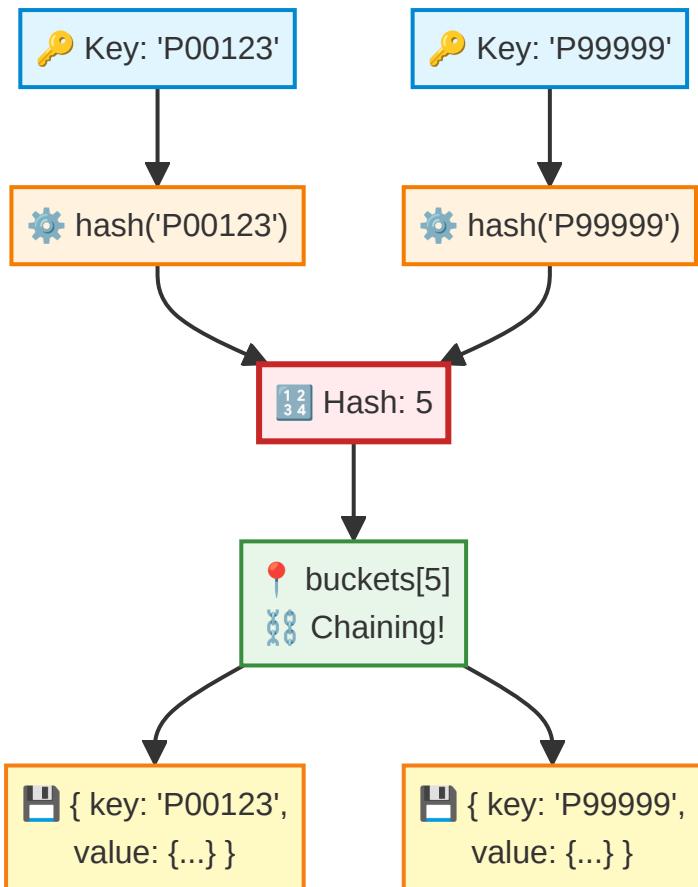
SET "P00567" → Hash: 8
SET "P00123" → Hash: 6
SET "P00999" → Hash: 7

--- Lookups ---
GET "P00567" → Hash: 8
Found: {"name":"Laptop","price":999}
GET "P00123" → Hash: 6
Found: {"name":"Mouse","price":25}
GET "P99999" → Hash: 5
Found: undefined

```

Seht ihr? Die Hash-Funktion berechnet aus dem Key einen Index. Dann greifen wir direkt auf **buckets[index]** zu –  $O(1)$ ! Das ist die Magie hinter Hash-Tables.

Problem: Hash-Kollisionen



Hash-Kollisionen sind unvermeidbar! Die Lösung: Chaining. Jeder Bucket speichert eine Liste von Key-Value-Paaren. Bei einer Kollision durchsuchen wir nur diese kleine Liste – immer noch sehr schnell!

#### Verbesserte Implementation mit Collision-Handling

```

1 * class HashTableWithChaining {
2 *   constructor(size = 10) {
3 *     this.buckets = new Array(size).fill(null).map(() => []);
4 *     this.size = size;
5 *   }
6 *
7 *   hash(key) {
8 *     let hashValue = 0;
9 *     for (let i = 0; i < key.length; i++) {
10 *       hashValue += key.charCodeAt(i);
11 *     }
12 *     return hashValue % this.size;
13 *   }
14 *
15 *   set(key, value) {
16 *     const index = this.hash(key);
17 *     const bucket = this.buckets[index];
18 *
19 *     // Prüfen ob Key schon existiert (Update)
20 *     for (let i = 0; i < bucket.length; i++) {
21 *       if (bucket[i].key === key) {

```

```

22     bucket[i].value = value;
23     console.log(`UPDATE "${key}" at bucket ${index}`);
24     return;
25   }
26 }
27
28 // Neuer Key → anhängen
29 bucket.push({ key, value });
30 console.log(`INSERT "${key}" at bucket ${index} (chain length: ${
31   .length})`);
32 }
33
34 get(key) {
35   const index = this.hash(key);
36   const bucket = this.buckets[index];
37
38   // Bucket durchsuchen
39   for (let i = 0; i < bucket.length; i++) {
40     if (bucket[i].key === key) {
41       console.log(`FOUND "${key}" at bucket ${index}`);
42       return bucket[i].value;
43     }
44   }
45
46   console.log(`NOT FOUND "${key}"`);
47   return undefined;
48 }
49
50 // Debugging: Alle Buckets anzeigen
51 showBuckets() {
52   console.log("\n==== Hash Table Structure ====");
53   for (let i = 0; i < this.size; i++) {
54     if (this.buckets[i].length > 0) {
55       const keys = this.buckets[i].map(item => item.key).join(", ")
56       console.log(`Bucket ${i}: [${keys}]`);
57     }
58   }
59 }
60
61 // Demo mit Kollisionen
62 const ht = new HashTableWithChaining(5); // Kleiner = mehr Kollision
63
64 ht.set("P00123", { name: "Mouse", price: 25 });
65 ht.set("P00567", { name: "Laptop", price: 999 });
66 ht.set("P99999", { name: "Monitor", price: 299 });
67 ht.set("P00321", { name: "Keyboard", price: 75 });
68
69 ht.showBuckets();
70
71 console.log("\n==== Lookups ====");

```

```

1  console.log(`--- Lookups ---`);  

2  console.log("Mouse:", ht.get("P00123"));  

3  console.log("Laptop:", ht.get("P00567"));  

4  console.log("Unknown:", ht.get("XXXXX"));

```

```

INSERT "P00123" at bucket 1 (chain length: 1)  

INSERT "P00567" at bucket 3 (chain length: 1)  

INSERT "P99999" at bucket 0 (chain length: 1)  

INSERT "P00321" at bucket 1 (chain length: 2)

=== Hash Table Structure ===  

Bucket 0: [P99999]  

Bucket 1: [P00123, P00321]  

Bucket 3: [P00567]

--- Lookups ---  

FOUND "P00123" at bucket 1  

Mouse: {"name": "Mouse", "price": 25}  

FOUND "P00567" at bucket 3  

Laptop: {"name": "Laptop", "price": 999}  

NOT FOUND "XXXXX"  

Unknown: undefined

```

Diese Implementation zeigt: Selbst mit Kollisionen ist der Lookup extrem schnell! Solange die Bucket-Listen kurz bleiben (gute Hash-Funktion + ausreichende Größe), haben wir praktisch  $O(1)O(1)$ .

**In der Praxis:** JavaScript-Objekte, Redis, Python Dictionaries – alle nutzen Hash-Tables mit ausgeklügelten Hash-Funktionen und automatischer Größenanpassung (Rehashing).

## Grenzen von Key-Value Stores

Jetzt kommt der Haken: Key-Value ist perfekt für ID-Lookups, aber was ist mit komplexeren Abfragen?

**Szenario:** Finde alle Produkte mit Lagerbestand < 10 (Low Stock Alert)

```

1  function filterProductsBy(filter) {  

2      const start = performance.now();  

3      const results = [];  

4  

5      // Müssen ALLE Keys durchsuchen! (wieder  $O(n)$ )  

6      for (const productId in db) {  

7          sleep(1); // Simuliere Latenz  

8          const product = db[productId];  

9          if (filter(product)) {  

10             results.push(product);  

11         }

```

```
12  }
13
14  const end = performance.now();
15
16  if (results.length > 0) {
17      console.log("✅ Produkte gefunden:", JSON.stringify(results, null));
18  } else {
19      console.log("❌ Keine Produkte gefunden");
20  }
21
22  console.log("⌚ Zeit:", (end - start).toFixed(2), "ms");
23 }
24
25 filterProductsBy(p => parseInt(p.stock) < 10);
```

 Produkte gefunden: [

```
{  
    "product_id": "P00011",  
    "name": "4K Monitor",  
    "category": "Electronics",  
    "brand": "Neutrino",  
    "price": "24.92",  
    "stock": "1",  
    "rating": "3.3",  
    "created_at": "2025-07-13"  
},  
{  
    "product_id": "P00065",  
    "name": "Gaming Mouse Plus S",  
    "category": "Electronics",  
    "brand": "BlueWave",  
    "price": "562.30",  
    "stock": "5",  
    "rating": "4.0",  
    "created_at": "2025-07-05"  
},  
{  
    "product_id": "P00085",  
    "name": "Portable SSD",  
    "category": "Electronics",  
    "brand": "Neutrino",  
    "price": "968.47",  
    "stock": "4",  
    "rating": "3.2",  
    "created_at": "2024-06-10"  
},  
{  
    "product_id": "P00086",  
    "name": "Tablet 10\" Pro - Black",  
    "category": "Electronics",  
    "brand": "Voltix",  
    "price": "827.84",  
    "stock": "3",  
    "rating": "3.4",  
    "created_at": "2025-10-10"  
},  
{  
    "product_id": "P00096",  
}
```

```
"name": "Bluetooth Speaker",
"category": "Electronics",
"brand": "ZenCore",
"price": "1493.82",
"stock": "5",
"rating": "4.0",
"created_at": "2024-09-19"
},
{
"product_id": "P00132",
"name": "Dress Shirt Plus - Graphite",
"category": "Clothing",
"brand": "PureThread",
"price": "198.67",
"stock": "3",
"rating": "2.4",
"created_at": "2025-02-17"
},
{
"product_id": "P00141",
"name": "Dress Shirt - Blue",
"category": "Clothing",
"brand": "AeroLoom",
"price": "39.97",
"stock": "3",
"rating": "4.9",
"created_at": "2025-03-08"
},
{
"product_id": "P00393",
"name": "Whiteboard Markers (8-pack)",
"category": "Office",
"brand": "InkFlow",
"price": "415.87",
"stock": "2",
"rating": "3.4",
"created_at": "2025-05-16"
},
{
"product_id": "P00416",
"name": "Cutlery Set (24 pcs) - Graphite",
"category": "Home & Kitchen",
"brand": "Hearthstone",
```

```
"price": "130.55",
"stock": "7",
"rating": "3.3",
"created_at": "2025-10-18"
},
{
  "product_id": "P00450",
  "name": "Toaster 2-slice Max L",
  "category": "Home & Kitchen",
  "brand": "CozyNest",
  "price": "148.36",
  "stock": "3",
  "rating": "3.9",
  "created_at": "2025-01-19"
},
{
  "product_id": "P00749",
  "name": "Micellar Water 400ml Pro",
  "category": "Beauty",
  "brand": "AurumCare",
  "price": "100.06",
  "stock": "7",
  "rating": "4.3",
  "created_at": "2024-10-24"
},
{
  "product_id": "P00804",
  "name": "Poetry Collection Pro - White",
  "category": "Books",
  "brand": "Silver Quill",
  "price": "27.97",
  "stock": "2",
  "rating": "3.3",
  "created_at": "2025-03-09"
},
{
  "product_id": "P00809",
  "name": "Beginners German Max",
  "category": "Books",
  "brand": "Atlas Editions",
  "price": "51.71",
  "stock": "9",
  "rating": "4.7",
```

```

    "created_at": "2024-07-27"
},
{
  "product_id": "P00854",
  "name": "Poetry Collection",
  "category": "Books",
  "brand": "Silver Quill",
  "price": "70.25",
  "stock": "3",
  "rating": "4.0",
  "created_at": "2024-05-05"
},
{
  "product_id": "P00878",
  "name": "Cookbook - Quick Meals 2.0",
  "category": "Books",
  "brand": "Northwind Press",
  "price": "53.36",
  "stock": "3",
  "rating": "4.8",
  "created_at": "2024-12-28"
},
{
  "product_id": "P00902",
  "name": "Bird Seed 1kg Max",
  "category": "Pets",
  "brand": "WildFeast",
  "price": "93.18",
  "stock": "1",
  "rating": "4.3",
  "created_at": "2025-01-19"
}
]
⌚ Zeit: 1003.20 ms

```

Und schon sind wir wieder bei  $O(n)$ ! Key-Value ist fantastisch für exakte ID-Lookups, aber für Range-Queries oder Filter brauchen wir andere Paradigmen. Das ist der Trade-off!

**Key Insight:** Key-Value ist kein Allheilmittel – es ist ein Werkzeug für einen spezifischen Use Case:  
schneller Key-basierter Zugriff.

## Das Key-Value Prinzip: Formalisiert

Gut! Wir haben gesehen, dass Hash-basierte Indizes blitzschnell sind. Jetzt formalisieren wir das Ganze: Was genau ist ein Key-Value Store?

## Definition

Ein Key-Value Store ist eine Datenstruktur, die:

- **Keys** (eindeutige Identifikatoren, meist Strings) auf **Values** (beliebige Daten) abbildet
- $O(1)$  Lookup garantiert (durchschnittlicher Fall)
- **Keine Struktur** für Values vorgibt (Blobs, JSON, Strings, ...)
- **Kein Schema** erzwingt

Heute schauen wir uns an, wie professionelle Key-Value Stores wie Redis diese Prinzipien umsetzen – und welche zusätzlichen Features sie bieten.

## Referenzen

- Redis Documentation: [Commands Overview](#)
- MDN Web Docs: [Web Storage API](#)

# Historische Entwicklung der Key-Value Stores

Bevor wir die technischen Details betrachten, lohnt sich ein Blick in die Geschichte: Wie entstand das Key-Value-Prinzip und warum ist es heute so wichtig?

## Frühe Anfänge

- Schon in den 1960er/70er Jahren tauchten assoziative Arrays in Sprachen wie Lisp und Perl auf.
- Betriebssysteme nutzten Key-Value-Konzepte für Konfigurationsdateien und Registries.

Die Idee, Daten über eindeutige Schlüssel blitzschnell zu finden, ist also älter als die meisten modernen Datenbanksysteme.

## Erste Datenbanksysteme & Web-Ära

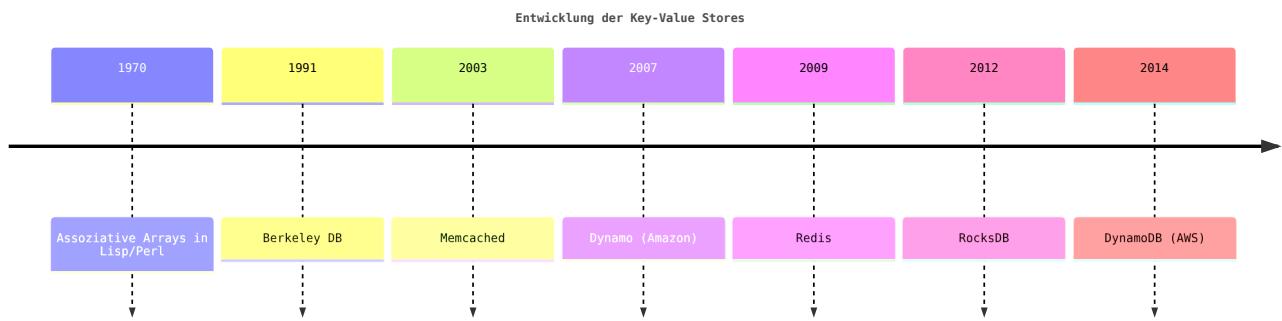
- Mit Berkeley DB und später Memcached (Ende 1990er/2000er) wurde das Prinzip für Performance und horizontale Skalierung genutzt.
- Die Web-Ära brachte neue Herausforderungen: Millionen von Nutzern, Sessions, Caching – Key-Value Stores wurden zum Rückgrat vieler Webanwendungen.

Das CAP-Theorem (Consistency, Availability, Partition Tolerance) trieb die Entwicklung von verteilten Key-Value Stores wie DynamoDB und Redis voran.

## Moderne Key-Value Stores

- Heute sind Redis, DynamoDB, RocksDB und LevelDB die Platzhirsche.
- Features wie TTL, atomare Operationen, Replikation und Sharding machen sie unverzichtbar für Cloud, Microservices und Echtzeit-Anwendungen.

Key-Value Stores sind nicht nur schnell, sondern auch extrem flexibel und skalierbar.



Der Zeitstrahl zeigt: Die Entwicklung verlief parallel zu den Anforderungen an Skalierbarkeit und Performance im Web.

## Visualisierung: Key → Value Mapping

Das Grundprinzip ist denkbar einfach: Jeder Wert wird über einen eindeutigen Schlüssel gespeichert und abgerufen. Kein Schema, keine Tabellen, keine Joins – nur Schlüssel und Werte.

Key–Value Store	
'user:42'	→ { name: 'Alice' , . . . }
'session:xyz'	→ { expires: 1700000000 }
'counter:views'	→ 12345

Der Clou: Der Zugriff erfolgt in konstanter Zeit – O von 1. Egal ob ihr 100 oder 100 Millionen Keys habt, der Lookup ist gleich schnell. Das ist die Superkraft von Key-Value Stores.

### Grundoperationen

Operation	Beschreibung	Beispiel
SET	Wert speichern	SET user:42 '{"name":"Alice"}'
GET	Wert abrufen	GET user:42
DELETE	Wert löschen	DELETE user:42
EXISTS	Prüfen ob Key existiert	EXISTS user:42

Diese vier Operationen sind das Rückgrat jedes Key-Value Stores. Simpel, aber mächtig.

## Redis in Action

Schauen wir uns das Ganze in Aktion an. Ich zeige euch echte Redis-Operationen, die ihr direkt im Browser ausführen könnt – ohne Server-Setup!

### Redis Basics: SET & GET

```
1 const redis = new Redis();
2
3 // User-Daten speichern
4 await redis.set('user:42', JSON.stringify({
5   name: "Alice",
6   email: "alice@example.org"
7 }));
8
9 // Daten abrufen
10 const userData = await redis.get('user:42');
11 const user = JSON.parse(userData);
12
13 console.log('User Name:', user.name);
14 console.log('User Email:', user.email);
15
16 // Atomic Counter
17 await redis.set('page:views', '0');
18 await redis.incr('page:views');
19 await redis.incr('page:views');
20 await redis.incr('page:views');
21
22 const views = await redis.get('page:views');
23 console.log('Page Views:', views);
```

```
User Name: Alice
User Email: alice@example.org
Page Views: 3
```

Redis ist der Klassiker: In-Memory, blitzschnell, mit optionaler Persistierung. Perfekt für Caching, Session Stores, Leaderboards. Hier läuft es direkt im Browser – probiert es aus!

### Hash Operations: Strukturierte Objekte

```
1 const redis = new Redis();
2
3 // User als Hash speichern (effizienter!)
4 await redis.hset('user:1001', 'name', 'Bob');
5 await redis.hset('user:1001', 'email', 'bob@example.org');
6 await redis.hset('user:1001', 'age', '28');
7 await redis.hset('user:1001', 'role', 'developer');
```

```

8 // Gesamtes Hash abrufen
9
10 const user = await redis.hgetall('user:1001');
11 console.log('User Object:', user);
12
13 // Einzelnes Feld abrufen
14 const name = await redis.hget('user:1001', 'name');
15 console.log('Name:', name);
16
17 // Feld existiert?
18 const hasPhone = await redis.exists('user:1001', 'phone');
19 console.log('Has Phone:', hasPhone === 1);

```

User Object:

{"name": "Bob", "email": "bob@example.org", "age": "28", "role": "developer"}

Name: Bob

Has Phone: false

Redis Hashes sind effizienter als JSON-Strings, wenn ihr nur einzelne Felder abrufen oder ändern wollt.  
Jedes Feld ist separat zugreifbar – perfekt für User-Profile!

## TTL & Expiration: Selbstlöschende Daten

Eine der Killer-Features von Key-Value Stores: Time-To-Live oder TTL. Keys können automatisch ablaufen – perfekt für Sessions, Caches, Rate Limiting.

### TTL in Action

```

1 const redis = new Redis();
2
3 // Session mit 10 Sekunden TTL
4 await redis.set('session:abc', JSON.stringify({ user_id: 42 }), 'EX',
5
6 // TTL abfragen
7 let ttl1 = await redis.ttl('session:abc');
8 console.log('TTL session:abc:', ttl1, 'seconds');
9
10 // Alternative: TTL nachträglich setzen
11 await redis.set('cache:data', 'cached_value');
12 await redis.expire('cache:data', 30);
13
14 let ttl2 = await redis.ttl('cache:data');
15 console.log('TTL cache:data:', ttl2, 'seconds');
16
17 // Key existiert?
18 const exists = await redis.exists('session:abc');
19 console.log('Session exists:', exists === 1);

```

```
TTL session:abc: 10 seconds
TTL cache:data: 30 seconds
Session exists: true
```

Nach Ablauf der Zeit ist der Key einfach weg. Keine manuellen Cleanup-Jobs nötig – der Store kümmert sich selbst darum. In diesem Beispiel sieht ihr, wie TTL funktioniert.

**Use Case:** Session Store – nach 1 Stunde Inaktivität wird die Session automatisch gelöscht. Perfekt für Web-Apps!

## Atomic Operations: Race Conditions vermeiden

Jetzt wird's interessant: Was passiert, wenn zwei Prozesse gleichzeitig denselben Counter hochzählen wollen?

**Das Problem: Lost Update**

```
1 // Simulation: Zwei "Prozesse" versuchen gleichzeitig zu inkrementieren
2 const redis = new Redis();
3 await redis.set('counter', '10');
4
5 // Naiver Ansatz (UNSICHER!)
6 async function unsafeIncrement(processId) {
7   const value = parseInt(await redis.get('counter'));
8   const newValue = value + 1;
9   await redis.set('counter', newValue.toString());
10
11   console.debug(`Process ${processId} increments counter to:`, newValue);
12
13   return newValue;
14 }
15
16 // Beide "Prozesse" lesen denselben Wert (10)
17 console.log('Process 1 reads:', await redis.get('counter'));
18 console.log('Process 2 reads:', await redis.get('counter'));
19
20 // Beide schreiben 11 - ein Update geht verloren!
21 await Promise.all([unsafeIncrement(1), unsafeIncrement(2)]);
22
23 const final = await redis.get('counter');
24 console.log('Final value:', final, '(should be 12, but is 11!)');
```

```
Process 1 reads: 10
Process 2 reads: 10
Process 1 increments counter to: 11
Process 2 increments counter to: 11
Final value: 11 (should be 12, but is 11!)
```

Das ist eine klassische Race Condition: Beide lesen denselben Wert, beide schreiben denselben Wert plus 1 – ein Inkrement geht verloren. Lost Update!

### Die Lösung: Atomic INCR

```
1 const redis = new Redis();
2 await redis.set('counter', '10');
3
4 // Atomares Inkrementieren (SICHER!)
5 const result1 = await redis.incr('counter');
6 console.log('After INCR #1:', result1);
7
8 const result2 = await redis.incr('counter');
9 console.log('After INCR #2:', result2);
10
11 const result3 = await redis.incr('counter');
12 console.log('After INCR #3:', result3);
13
14 // Auch mit INCRBY
15 await redis.incrby('counter', 5);
16 const final = await redis.get('counter');
17 console.log('After INCRBY 5:', final);
```

```
After INCR #1: 11
After INCR #2: 12
After INCR #3: 13
After INCRBY 5: 18
```

Redis garantiert: INCR ist atomar. Mehrere Prozesse können parallel inkrementieren, ohne dass Updates verloren gehen. Das ist die Magie atomarer Operationen!

### Weitere atomare Operationen

Operation	Beschreibung
INCR	Atomares Inkrementieren um 1
DECR	Atomares Dekrementieren um 1
INCRBY n	Atomares Addieren von n
DECRBY n	Atomares Subtrahieren von n
GETSET	Setzen und alten Wert zurückgeben (atomar)
SETEX	Set if Not eXists (für Locking/Semaphoren)

Diese Operationen sind eure Waffen gegen Race Conditions. Nutzt sie!

## Key-Design Patterns: Organisation ist alles

Keys sind nur Strings – aber wie organisiert man tausende oder Millionen Keys sinnvoll? Hier kommen Key-Design Patterns ins Spiel.

### Pattern 1: Namespacing

```
user:42:profile
user:42:settings
user:42:sessions

product:99:details
product:99:reviews
product:99:inventory
```

Mit Doppelpunkten strukturieren wir Keys hierarchisch. Das ist nur eine Konvention, aber sie macht Keys lesbar und wartbar.

### Pattern 2: Composite Keys

```
session:{user_id}:{timestamp}
cart:{session_id}:{product_id}
rate_limit:{ip}:{endpoint}:{minute}
```

Mehrere Dimensionen in einem Key – perfekt für komplexe Zugriffsszenarien.

### Pattern 3: Leaderboard mit Sorted Sets

```
1 const redis = new Redis();
2
```

```

3 // Gaming Leaderboard erstellen
4 await redis.zadd('leaderboard', 9500, 'player:alice');
5 await redis.zadd('leaderboard', 8200, 'player:bob');
6 await redis.zadd('leaderboard', 12000, 'player:charlie');
7 await redis.zadd('leaderboard', 7800, 'player:diana');
8 await redis.zadd('leaderboard', 10500, 'player:eve');
9
10 // Top 3 Spieler (absteigend sortiert)
11 let topPlayers = await redis.zrevrange('leaderboard', 0, 2, 'WITHSCORES');
12 console.log(`🏆 Top 3 Players:`);
13 for (let i = 0; i < topPlayers.length; i += 2) {
14   const rank = i / 2 + 1;
15   const player = topPlayers[i];
16   const score = topPlayers[i + 1];
17   console.log(` ${rank}. ${player} - ${score} points`);
18 }
19
20 // Rang eines bestimmten Spielers
21 const aliceRank = await redis.zrevrank('leaderboard', 'player:alice')
22 console.log(`\nAlice's Rank: #${aliceRank + 1}`);
23
24 // Score erhöhen (atomar!)
25 await redis.zincrby('leaderboard', 500, 'player:alice');
26 const newScore = await redis.zscore('leaderboard', 'player:alice');
27 console.log(`Alice's new score: ${newScore}`);

```

🏆 Top 3 Players:

1. player:charlie - 12000 points
2. player:eve - 10500 points
3. player:alice - 9500 points

Alice's Rank: #3

Alice's new score: 10000

Redis Sorted Sets sind perfekt für Ranglisten, Top-N Queries, Zeitreihen. Jedes Element hat einen Score, die Sortierung erfolgt automatisch. Ideal für Gaming, Analytics, Trending Topics!

## Referenzen

- Key Design: [Redis Best Practices](#)
- Sorted Sets: [ZADD Command](#)

## Typische Use Cases



## Wo KV glänzt

Jetzt wissen wir, WIE Key-Value funktioniert. Aber WANN sollten wir es einsetzen?

### Use Case Matrix

Use Case	Warum KV?	Beispiel
Caching	$O(1)$ Lookup, TTL	Seiten-Cache, API-Responses
Session Store	Schneller Zugriff, Expiration	User Sessions, JWT
Rate Limiting	Atomic Counters, TTL	API Rate Limits (pro Minute)
Leaderboards	Sorted Sets	Gaming, Analytics
Feature Flags	Schneller Read, einfaches Update	A/B Tests, Rollouts

Überall dort, wo ihr schnellen Key-basierten Zugriff braucht und keine komplexen Queries, ist Key-Value ideal.

### Referenzen

- Caching Strategies: [Cache Stampede Problem](#)
- Rate Limiting: [Token Bucket Algorithm](#)

# Grenzen: Was KV NICHT kann

Kein Paradigma ist perfekt. Schauen wir uns an, wo Key-Value an seine Grenzen stößt.

## Problem 1: Keine Range Queries

```
// Ich will alle User mit IDs zwischen 100 und 200
// ❌ Nicht möglich mit Standard Key-Value!
// Workaround: Alle Keys durchsuchen (teuer!)
KEYS user:*
```



Key-Value kennt nur exakte Lookups. Bereichsabfragen? Fehlanzeige. Dafür bräuchten wir Indexstrukturen – die hat KV nicht.

## Problem 2: Keine Joins

```
// Ich will alle Bestellungen eines Users UND die Produktdetails
// ❌ Keine Joins in KV!
// Workaround: Mehrere GET Requests
GET user:42:orders    // → [99, 100, 101]
GET product:99
GET product:100
GET product:101
```



Jede Relation muss manuell aufgelöst werden. Bei komplexen Datenmodellen wird das schnell unübersichtlich.

## Problem 3: Keine Ad-hoc Analysen

```
// "Zeige alle User, die in den letzten 7 Tagen aktiv waren"
// ❌ Ohne Secondary Indexes nicht möglich
```



Analytische Abfragen erfordern Strukturinformationen, die Key-Value nicht hat. Dafür brauchen wir Document, Column oder Relational Stores.

**Trade-off:** KV ist simpel und schnell, aber unflexibel. Für komplexe Queries brauchen wir reichhaltigere Paradigmen.

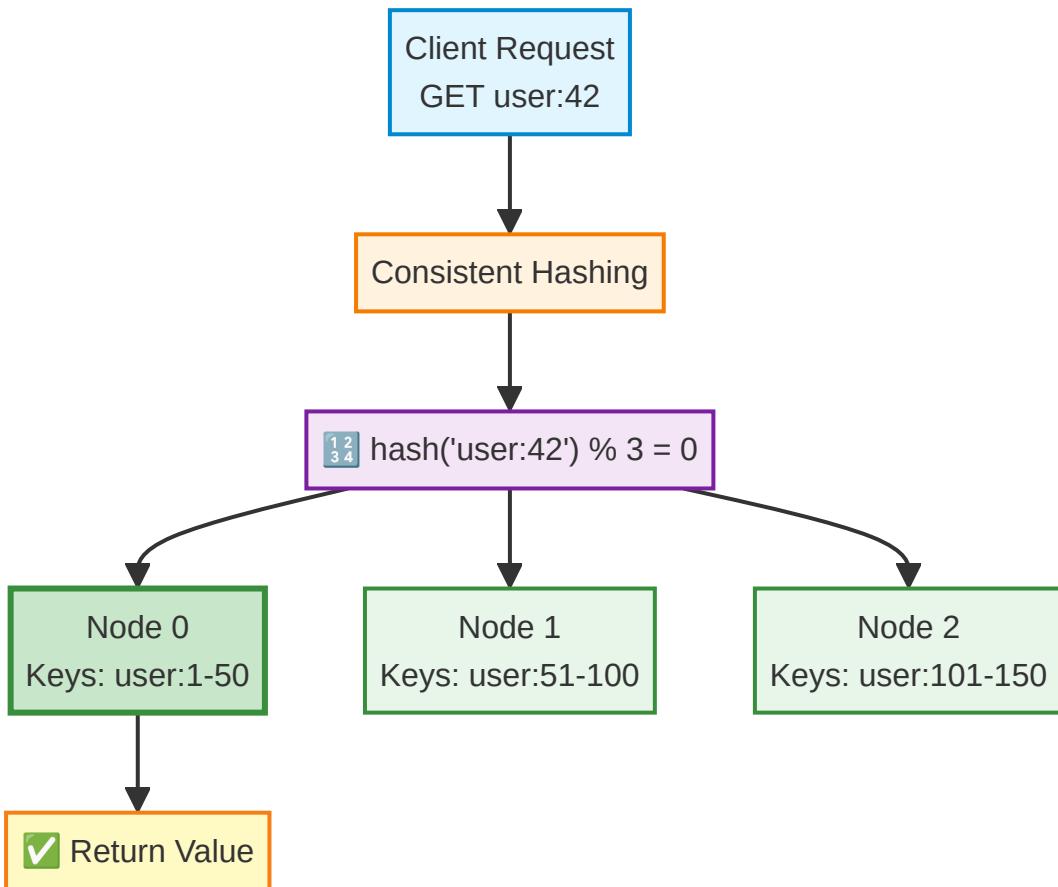
# Parallelisierung und Skalierung

Ein letzter Punkt: Wie skaliert Key-Value, wenn die Datenmengen wachsen? Key-Value Stores haben hier einen entscheidenden Vorteil: Sie sind von Natur aus horizontal skalierbar.

# Sharding: Daten auf mehrere Knoten verteilen

Sharding bedeutet, dass wir unsere Keys auf mehrere Nodes verteilen. Jeder Node ist für einen bestimmten Key-Bereich verantwortlich.

Visualisierung: Hash-based Sharding



Mit Consistent Hashing berechnen wir aus jedem Key, auf welchem Node die Daten liegen. Das ermöglicht horizontale Skalierung: Mehr Nodes = mehr Kapazität!

Beispiel: Sharding in der Praxis

```
1 // Simulation eines einfachen Sharding-Clients
2 class ShardedKeyValueClient {
3   constructor(nodes) {
4     this.nodes = nodes; // Array von Node-IDs
5   }
6
7 // Consistent Hashing: Key → Node
8 getNodeForKey(key) {
9   let hash = 0;
10  for (let i = 0; i < key.length; i++) {
11    hash += key.charCodeAt(i);
12  }
13  const nodeIndex = hash % this.nodes.length;
14  return this.nodes[nodeIndex];
```

```

15    }
16
17    // GET: Ermittle Node und hole Wert
18    get(key) {
19        const node = this.getNodeForKey(key);
20        console.log(`GET "${key}" → routed to ${node}`);
21        // In der Realität: HTTP Request an den Node
22        return `[Value from ${node}]`;
23    }
24
25    // SET: Ermittle Node und speichere Wert
26    set(key, value) {
27        const node = this.getNodeForKey(key);
28        console.log(`SET "${key}" → routed to ${node}`);
29        // In der Realität: HTTP Request an den Node
30        return `OK (stored on ${node})`;
31    }
32 }
33
34 // Demo: 3 Nodes
35 const client = new ShardedKeyValueClient(['node-0', 'node-1', 'node-2'
36
37 console.log('== Sharding Demo ==\n');
38
39 // Verschiedene Keys werden auf unterschiedliche Nodes verteilt
40 client.set('user:42', '{"name":"Alice"}');
41 client.set('user:99', '{"name":"Bob"}');
42 client.set('product:123', '{"name":"Laptop"}');
43 client.set('session:abc', '{"user_id":42}');
44 client.set('cart:xyz', '{"items":[1,2,3]}');
45
46 console.log('\n--- Retrieving Data ---\n');
47
48 client.get('user:42');
49 client.get('product:123');
50 client.get('session:abc');
51
52 console.log(`\n💡 Jeder Key landet auf einem bestimmten Node!`);
53 console.log('→ Mehr Nodes = mehr Parallelität und Kapazität');

```

```
==== Sharding Demo ====
```

```
SET "user:42" → routed to node-1  
SET "user:99" → routed to node-1  
SET "product:123" → routed to node-2  
SET "session:abc" → routed to node-2  
SET "cart:xyz" → routed to node-1
```

```
==== Retrieving Data ====
```

```
GET "user:42" → routed to node-1  
GET "product:123" → routed to node-2  
GET "session:abc" → routed to node-2
```

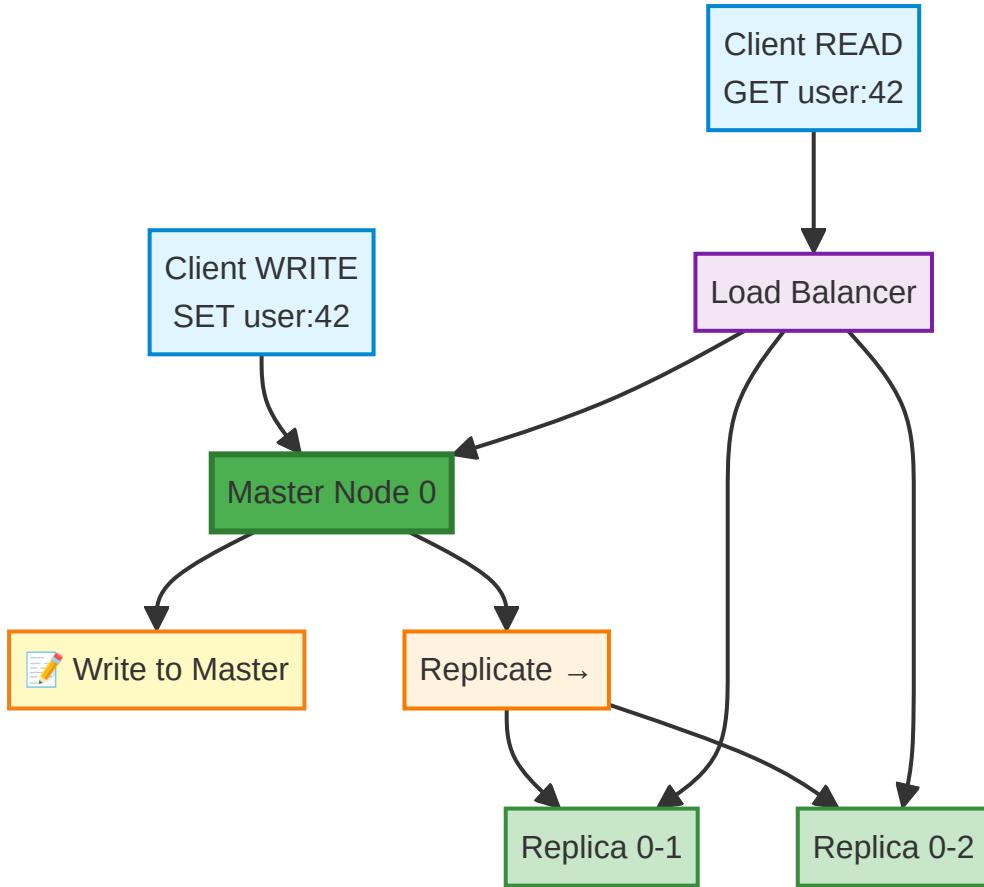
 Jeder Key landet auf einem bestimmten Node!  
→ Mehr Nodes = mehr Parallelität und Kapazität

Seht ihr? Jeder Key wird automatisch einem Node zugeordnet. Das System skaliert linear: 3 Nodes können 3x so viele Requests parallel verarbeiten wie 1 Node!

## Replication: Ausfallsicherheit durch Redundanz

Sharding allein reicht nicht – was passiert, wenn ein Node ausfällt? Hier kommt Replication ins Spiel.

Master-Replica Architektur



Schreiboperationen gehen an den Master, der dann asynchron an Replicas repliziert. Lesezugriffe können vom Master oder von beliebigen Replicas bedient werden – das verteilt die Last!

#### Trade-offs bei Replication

Replikations-Typ	Konsistenz	Performance	Use Case
Synchron	✓ Stark	⚠ Langsam	Finanztransaktionen
Asynchron	⚠ Eventual	✓ Schnell	Social Media Feeds, Caching
Semi-Synchron	🟡 Mittel	🟡 Mittel	E-Commerce, Session Stores

Der Klassiker: Konsistenz vs. Performance. Für Session Stores ist Eventual Consistency meist ok – für Banküberweisungen nicht!

## Redis Enterprise & CRDTs

Jetzt wird's spannend: Redis Enterprise nutzt **CRDTs** (Conflict-free Replicated Data Types) für Multi-Master-Replication ohne Konflikte!

Was sind CRDTs?

CRDTs sind Datenstrukturen, die **garantiert konfliktfrei** zusammengeführt werden können, selbst wenn mehrere Replicas gleichzeitig Updates erhalten.

### Beispiel: Counter-CRDT

```
1 // Klassisches Problem: Zwei Replicas inkrementieren gleichzeitig
2 // Replica 1: counter = 10 → 11
3 // Replica 2: counter = 10 → 11
4 // Merge: Was ist das Ergebnis? 11? 12? ❌ Konflikt!
5
6 // CRDT-Lösung: G-Counter (Grow-only Counter)
7 class GCounter {
8   constructor(replicaId) {
9     this.replicaId = replicaId;
10    this.counters = {};// Pro Replica ein eigener Counter
11  }
12
13  increment() {
14    if (!this.counters[this.replicaId]) {
15      this.counters[this.replicaId] = 0;
16    }
17    this.counters[this.replicaId]++;
18    console.log(`Replica ${this.replicaId}: local counter = ${this
19      .counters[this.replicaId]}`);
20  }
21
22  value() {
23    // Gesamtwert: Summe aller Replica-Counter
24    return Object.values(this.counters).reduce((sum, val) => sum + va
25  }
26
27  merge(other) {
28    // Merge: Maximum pro Replica nehmen
29    for (const [replica, count] of Object.entries(other.counters)) {
30      if (!this.counters[replica] || this.counters[replica] < count)
31        this.counters[replica] = count;
32    }
33    console.log(`Merged: total = ${this.value()}`);
34  }
35}
36
37 // Simulation: Zwei Replicas inkrementieren unabhängig
38 console.log('== CRDT Counter Demo ==\n');
39
40 const replica1 = new GCounter('replica-1');
41 const replica2 = new GCounter('replica-2');
42
43 replica1.increment(); // replica-1: 1
44 replica1.increment(); // replica-1: 2
45
```

```

46 replica2.increment(); // replica-2: 1
47 replica2.increment(); // replica-2: 2
48 replica2.increment(); // replica-2: 3
49
50 console.log('\nBefore merge:');
51 console.log(` Replica 1 sees: ${replica1.value()}`);
52 console.log(` Replica 2 sees: ${replica2.value()}`);
53
54 console.log('\nMerging Replica 2 → Replica 1:');
55 replica1.merge(replica2);
56
57 console.log('\nAfter merge:');
58 console.log(` Total: ${replica1.value()} ✓ (2 + 3 = 5, kein Verlust)

```

==== CRDT Counter Demo ===

```

Replica replica-1: local counter = 1
Replica replica-1: local counter = 2
Replica replica-2: local counter = 1
Replica replica-2: local counter = 2
Replica replica-2: local counter = 3

```

Before merge:

```

Replica 1 sees: 2
Replica 2 sees: 3

```

Merging Replica 2 → Replica 1:

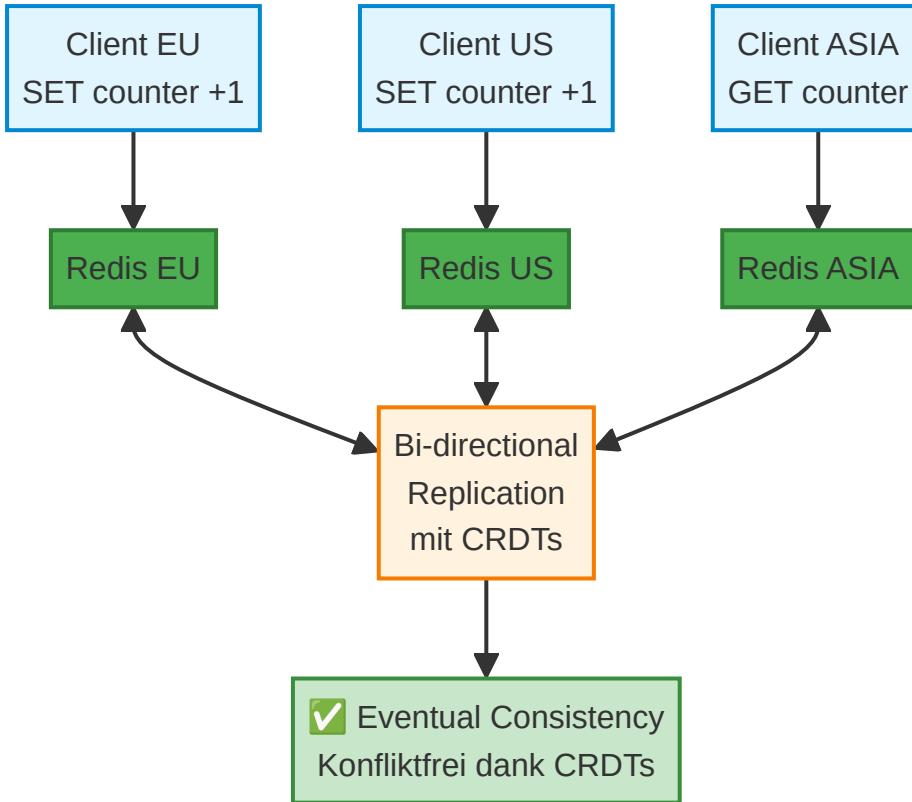
Merged: total = 5

After merge:

```
Total: 5 ✓ (2 + 3 = 5, kein Verlust!)
```

CRDTs lösen das Problem mathematisch: Jede Replica hat ihren eigenen Counter. Beim Merge nehmen wir das Maximum – garantiert konfliktfrei! Redis Enterprise nutzt dies für Active-Active Geo-Replication.

**Redis Enterprise: Active-Active mit CRDTs**



Mit Redis Enterprise könnt ihr weltweit verteilte Master-Nodes haben, die alle Writes akzeptieren. CRDTs garantieren, dass alle Replicas konfliktfrei konvergieren – ohne manuelle Konfliktauflösung!

#### CRDT-Typen in Redis Enterprise

CRDT-Typ	Operationen	Use Case
Counter	INCR, DECR	Page Views, Likes
Register	SET (Last-Write-Wins)	User Status, Config
Set	SADD, SREM	Tags, Permissions
Sorted Set	ZADD (max score wins)	Leaderboards
Hash	HSET (per-field LWW)	User Profiles

Redis Enterprise unterstützt verschiedene CRDT-Typen out-of-the-box. Ihr schreibt normalen Redis-Code, das System kümmert sich um Konfliktauflösung!

## Herausforderungen bei Skalierung

Trotz aller Vorteile gibt es Herausforderungen:

## Die Realität

- **Konsistenz:** Bei verteilten Systemen schwieriger zu gewährleisten
  - CAP-Theorem: Consistency, Availability, Partition Tolerance – pick two!
  - Eventual Consistency ist oft ein Kompromiss
- **Netzwerk-Latenz:** Mehr Nodes = mehr Kommunikation
  - Cross-Region Replication: 100+ ms Latenz
  - Lösung: Geo-Partitioning, lokale Caches
- **Rebalancing:** Bei Änderungen in der Datenverteilung aufwendig
  - Neuer Node hinzufügen? → Keys müssen umverteilt werden
  - Consistent Hashing minimiert Umverteilung (nur ~1/n der Keys bewegen sich)

Verteilte Systeme sind komplex! Redis Enterprise und ähnliche Lösungen nehmen euch viel Arbeit ab, aber das Verständnis der Trade-offs bleibt essentiell.

**Best Practice:** Startet einfach (Single Node), skaliert bei Bedarf horizontal. Premature Optimization ist die Wurzel allen Übels!

## Referenzen

- CRDTs Explained: [Conflict-free Replicated Data Types](#)
- Redis Enterprise: [Active-Active Geo-Distribution](#)
- CAP Theorem: [Brewer's Conjecture](#)
- Consistent Hashing: [Dynamo Paper \(Amazon\)](#)

## Zusammenfassung & Wrap-up

Fassen wir zusammen: Key-Value Stores sind das Schweizer Taschenmesser für schnelle, strukturierte Zugriffe. Simpel, robust, blitzschnell.

## Key Takeaways

- **✓ O(1) Lookup:** Konstante Zugriffszeit, egal bei welcher Datenmenge
- **✓ TTL & Expiration:** Selbstlöschende Daten für Sessions, Caches
- **✓ Atomare Operationen:** Race Conditions vermeiden mit `INCR`, `SETNX`
- **✓ Key-Design Patterns:** Namespacing, Composite Keys, Sorted Sets
- **✓ Replikation:** Daten werden auf mehrere Knoten verteilt (Master-Replica, semi-synchron, asynchron) für Ausfallsicherheit und Skalierung
- **✓ Partitionierung (Sharding):** Daten werden logisch oder technisch aufgeteilt, um große Datenmengen und viele Zugriffe zu bewältigen
- **✓ Konsistenzmodelle:** Von starker Konsistenz (alle sehen immer den aktuellen Stand) bis zu eventual consistency (Daten gleichen sich zeitverzögert ab)
- **✗ Keine Range Queries:** Nur exakte Lookups möglich
- **✗ Keine Joins:** Relationen manuell auflösen
- **✗ Keine Ad-hoc Analysen:** Strukturinformationen fehlen

Key-Value ist perfekt für Caching, Sessions, Rate Limiting – überall dort, wo Geschwindigkeit und Einfachheit zählen. Für komplexere Datenmodelle schauen wir uns in den nächsten Sessions Document und Column Stores an.

### Interaktives Redis-Terminal

```

1 // Redis Playground: Experimentiert selbst!
2 redis = new Redis();
3
4 // Initial Setup
5 await redis.set('counter', '0');
6 await redis.hset('user:demo', 'name', 'Demo User');
7 await redis.zadd('scores', 100, 'Alice');
8
9 console.log('🎮 Redis Playground Ready!\n');
10 console.log('Try these commands:');
11 console.log('  await redis.incr("counter")');
12 console.log('  await redis.get("counter")');
13 console.log('  await redis.hgetall("user:demo")');
14 console.log('  await redis.zadd("scores", 200, "Bob")');
15 console.log('  await redis.zrevrange("scores", 0, -1, "WITHSCORES")')
16 console.log('  await redis.keys("*")');
```

Nutzt das Terminal, um selbst zu experimentieren! Probiert verschiedene Befehle aus, baut eigene Use Cases – Learning by Doing!