

# VR Mainfranken Select Union

Mischfonds - ISIN: DE0008007543



## WERTENTWICKLUNG IN DER VERGANGENHEIT\*

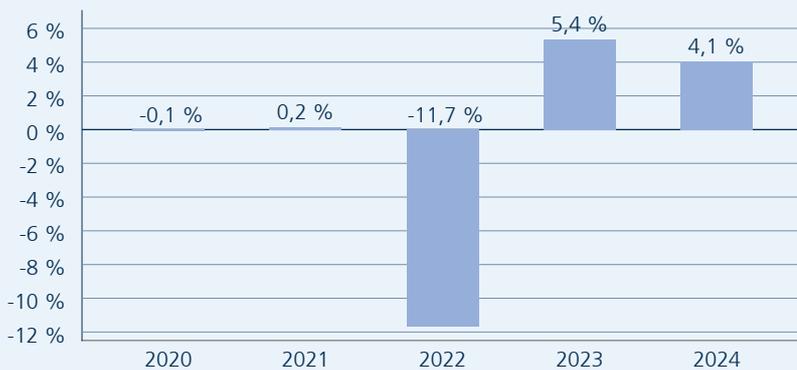
Indexierte Wertentwicklung



## WERTENTWICKLUNG ÜBER VERSCHIEDENE ZEITRÄUME

|              |              |               |                   |        |
|--------------|--------------|---------------|-------------------|--------|
| 1 Monat      | 3 Monate     | 6 Monate      | lfd. Jahr         | 1 Jahr |
| -0,27 %      | 2,38 %       | 9,34 %        | 4,07 %            | 7,81 % |
| 3 Jahre p.a. | 5 Jahre p.a. | 10 Jahre p.a. | Seit Auflage p.a. |        |
| -0,99 %      | -0,39 %      | 0,14 %        | 1,49 %            |        |
| 3 Jahre      | 5 Jahre      | 10 Jahre      | Seit Auflage      |        |
| -2,95 %      | -1,91 %      | 1,40 %        | 37,52 %           |        |

## WERTENTWICKLUNG PRO KALENDERJAHR



## ROLLIERENDE 12-MONATS ENTWICKLUNG

|                   |                   |                   |                   |
|-------------------|-------------------|-------------------|-------------------|
| 06.05.20-06.05.21 | 06.05.21-06.05.22 | 06.05.22-06.05.23 | 06.05.23-06.05.24 |
| 3,09 %            | -5,88 %           | -4,36 %           | 7,81 %            |

\*Historische Wertentwicklungen sind keine Garantie für eine ähnliche Entwicklung in der Zukunft. Diese ist nicht prognostizierbar. Die Berechnung der Wertentwicklung erfolgt gemäß BVI-Methode.

## 06. MAI 2024

### Anlageziel

Das Fondsvermögen wird in eine Vielzahl von Anlageklassen wie Aktien-, Renten-, Geldmarkt- sowie Alternative Investments investiert. Der Fonds legt als Mischfonds in Investmentfonds und Direktanlagen sowie in bereits getätigte Investitionen in offene Immobilienfonds an. Je nach Marktsituation wird das Fondsvermögen flexibel angelegt, was mit einem Wechsel von Anlageschwerpunkten einhergehen kann. Darüber hinaus wird überwiegend in Vermögensgegenstände investiert, deren Emittenten ethische, soziale und ökologische Kriterien berücksichtigen. Zur Umsetzung dieser nachhaltigen Anlagepolitik werden unter anderem Ausschlusskriterien festgelegt. Die Anlageentscheidung erfolgt aktiv nach einem strukturierten Investmentprozess. Mit dem Fonds werden darüber hinaus auch nachhaltige Investitionen gemäß Artikel 2 Ziffer 17 der Offenlegungsverordnung getätigt. Weiterhin werden beim Erwerb von Wertpapieren, Geldmarktinstrumenten und Investmentanteilen nachteilige Auswirkungen auf Nachhaltigkeitsfaktoren berücksichtigt. Die Anlagestrategie bezieht sich nicht auf einen Vergleichsmaßstab und verfolgt kein indexgebundenes Ziel. Der Fonds kann in den Raiffeisenbanken Bütthard-Gaukönigshofen eG, Höchberg eG, Mainschleife - Steigerwald eG, Main-Spessart eG, den VR-Banken Kitzingen eG, Lichtenfels-Ebern eG, Main-Rhön eG und Würzburg eG, den Raiffeisen-Volksbanken Aschaffenburg eG und Haßberge eG sowie der VR-MainBank eG, Vereinigte Volksbank Raiffeisenbank eG und der Frankfurter Volksbank Rhein/Main eG erworben werden.

```

class MirrorX(object):
    """This adds an
    """
    def __init__(self, x, y, z):
        self.x = x
        self.y = y
        self.z = z

    def __str__(self):
        return f"MirrorX({self.x}, {self.y}, {self.z})"

    def __repr__(self):
        return f"MirrorX({self.x}, {self.y}, {self.z})"

    def __eq__(self, other):
        return self.x == other.x and self.y == other.y and self.z == other.z

    def __neq__(self, other):
        return not self.__eq__(other)

    def __lt__(self, other):
        return self.x < other.x and self.y < other.y and self.z < other.z

    def __gt__(self, other):
        return self.x > other.x and self.y > other.y and self.z > other.z

    def __le__(self, other):
        return self.x <= other.x and self.y <= other.y and self.z <= other.z

    def __ge__(self, other):
        return self.x >= other.x and self.y >= other.y and self.z >= other.z

    def __add__(self, other):
        return MirrorX(self.x + other.x, self.y + other.y, self.z + other.z)

    def __sub__(self, other):
        return MirrorX(self.x - other.x, self.y - other.y, self.z - other.z)

    def __mul__(self, other):
        return MirrorX(self.x * other.x, self.y * other.y, self.z * other.z)

    def __div__(self, other):
        return MirrorX(self.x / other.x, self.y / other.y, self.z / other.z)

    def __mod__(self, other):
        return MirrorX(self.x % other.x, self.y % other.y, self.z % other.z)

    def __pow__(self, other):
        return MirrorX(self.x ** other.x, self.y ** other.y, self.z ** other.z)

    def __call__(self):
        return self

    def __getitem__(self, item):
        return self.__dict__[item]

    def __setitem__(self, key, value):
        self.__dict__[key] = value

    def __delitem__(self, key):
        del self.__dict__[key]

    def __contains__(self, item):
        return item in self.__dict__

    def __iter__(self):
        return iter(self.__dict__.values())

    def __len__(self):
        return len(self.__dict__)

    def __dir__(self):
        return dir(self.__dict__)

    def __hash__(self):
        return hash(self.x + self.y + self.z)

    def __bool__(self):
        return True

    def __float__(self):
        return float(self.x + self.y + self.z)

    def __int__(self):
        return int(self.x + self.y + self.z)

    def __complex__(self):
        return complex(self.x + self.y + self.z)

    def __bytes__(self):
        return bytes(self.x + self.y + self.z)

    def __str__(self):
        return f"MirrorX({self.x}, {self.y}, {self.z})"

    def __repr__(self):
        return f"MirrorX({self.x}, {self.y}, {self.z})"

    def __eq__(self, other):
        return self.x == other.x and self.y == other.y and self.z == other.z

    def __neq__(self, other):
        return not self.__eq__(other)

    def __lt__(self, other):
        return self.x < other.x and self.y < other.y and self.z < other.z

    def __gt__(self, other):
        return self.x > other.x and self.y > other.y and self.z > other.z

    def __le__(self, other):
        return self.x <= other.x and self.y <= other.y and self.z <= other.z

    def __ge__(self, other):
        return self.x >= other.x and self.y >= other.y and self.z >= other.z

    def __add__(self, other):
        return MirrorX(self.x + other.x, self.y + other.y, self.z + other.z)

    def __sub__(self, other):
        return MirrorX(self.x - other.x, self.y - other.y, self.z - other.z)

    def __mul__(self, other):
        return MirrorX(self.x * other.x, self.y * other.y, self.z * other.z)

    def __div__(self, other):
        return MirrorX(self.x / other.x, self.y / other.y, self.z / other.z)

    def __mod__(self, other):
        return MirrorX(self.x % other.x, self.y % other.y, self.z % other.z)

    def __pow__(self, other):
        return MirrorX(self.x ** other.x, self.y ** other.y, self.z ** other.z)

    def __call__(self):
        return self

    def __getitem__(self, item):
        return self.__dict__[item]

    def __setitem__(self, key, value):
        self.__dict__[key] = value

    def __delitem__(self, key):
        del self.__dict__[key]

    def __contains__(self, item):
        return item in self.__dict__

    def __iter__(self):
        return iter(self.__dict__.values())

    def __len__(self):
        return len(self.__dict__)

    def __dir__(self):
        return dir(self.__dict__)

    def __hash__(self):
        return hash(self.x + self.y + self.z)

    def __bool__(self):
        return True

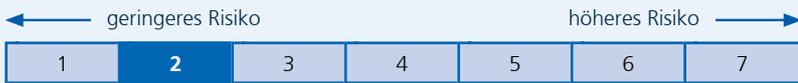
    def __float__(self):
        return float(self.x + self.y + self.z)

    def __int__(self):
        return int(self.x + self.y + self.z)

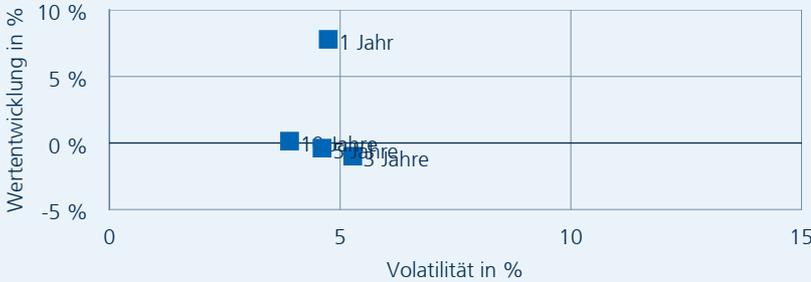
    def __complex__(self):
        return complex(self.x + self.y + self.z)

    def __bytes__(self):
        return bytes(self.x + self.y + self.z)
    
```

**RISIKOINDIKATOR\***



**Risiko-Rendite-Diagramm**



\*SRI gemäß Basisinformationsblatt

**PORTFOLIOSTRUKTUR**

**Top 5 Länderverteilung**



**Top 5 Währungsverteilung**



**Top 5 Holdings**



**Rechtliche Hinweise**

Datenquelle für Fondsinformationen: cleversoft GmbH. Die von der cleversoft GmbH bereitgestellten Informationen stellen keine Empfehlungen dar und dienen nicht der Anlageberatung. Insbesondere stellen diese Informationen keine Finanzanalyse im Sinne von § 34 b WpHG dar. Die cleversoft GmbH trifft keinerlei Aussage zur bisherigen und zukünftigen Wertentwicklung der angezeigten Wertpapiere. Wegen der Dynamik der Finanzmärkte muss jegliche Haftung im Zusammenhang mit der Nutzung der Informationen oder dem (evtl. auch irrtümlichen) Vertrauen auf deren Richtigkeit, Vollständigkeit oder Aktualität ausgeschlossen werden. Informieren Sie sich daher vor dem Fondserwerb auf der Internetseite der jeweiligen Fondsgesellschaft.

**Fonds-Fakten**

|                                       |   |
|---------------------------------------|---|
| Fondskategorie                        | Mischfonds                                      |
| WKN                                   | 800754  |
| ISIN                                  | DE0008007543                                    |
| Auflagedatum                          | 01. November 2002                               |
| Fondsvolumen                          | 257,33 Mio. EUR (06.05.2024)                    |
| Kapitalverwaltungs-gesellschaft       | Union Investment Privatfonds GmbH               |
| Fondsmanagement                       | Union Investment Gruppe                         |
| Depotbank                             | DZ Bank AG Deutsche Zentral-Genossenschaftsbank |
| Sitzland                              | Deutschland                                     |
| Fondswährung                          | EUR   |
| Rücknahmepreis                        | 56,29 EUR                                       |
| Ertragsverwendung                     | Ausschüttend                                    |
| Geschäftsjahr                         | 01. April - 31. März                            |
| Transparenzverord-nung (EU) 2019/2088 | Art. 8 TVO ESG Merkmale                         |

**Fonds-Konditionen**

|                                  |   |
|----------------------------------|---|
| Ausgabeauschlag                  | Ein Ausgabeaufschlag fällt im Rahmen der fondsgebundenen Rentenversicherungen von R+V nicht an. |
| Max. Verwaltungsvergütung p. a.  | 1,65 %  |
| Max. Fondsmanagement Gebühr p.a. | -   |
| Max. Depotbankver-gütung p.a.    | -   |
| Laufende Kosten p.a.             | 2,50 % (26.05.2023)   |
| Erfolgsabhängige Vergütung       | 0,00 %  |

**Kennzahlen**

| Volatilität  |         | Maximum Drawdown        |          |
|--------------|---------|-------------------------|----------|
| 1 Jahr       | +4,74 % | 1 Jahr                  | -4,89 %  |
| 3 Jahre      | +5,27 % | 3 Jahre                 | -14,48 % |
| 5 Jahre      | +4,61 % | 5 Jahre                 | -14,75 % |
| Sharpe Ratio |         | Verlustdauer in Monaten |          |
| 1 Jahr       | 0,83    | 1 Jahr                  | 3        |
| 3 Jahre      | -0,45   | 3 Jahre                 | 3        |
| 5 Jahre      | -0,22   | 5 Jahre                 | 3        |

Informationen erhalten Sie in den Volksbanken und Raiffeisenbanken, R+V-Agenturen sowie bei der Direktion der Gesellschaften der R+V Versicherungsgruppe, Raiffeisenplatz 1, 65189 Wiesbaden.

Telefon: 0800 533-1171

Kostenfrei aus allen deutschen Fest- und Mobilfunknetzen.

[www.ruv.de](http://www.ruv.de)

R+V Lebensversicherung AG