

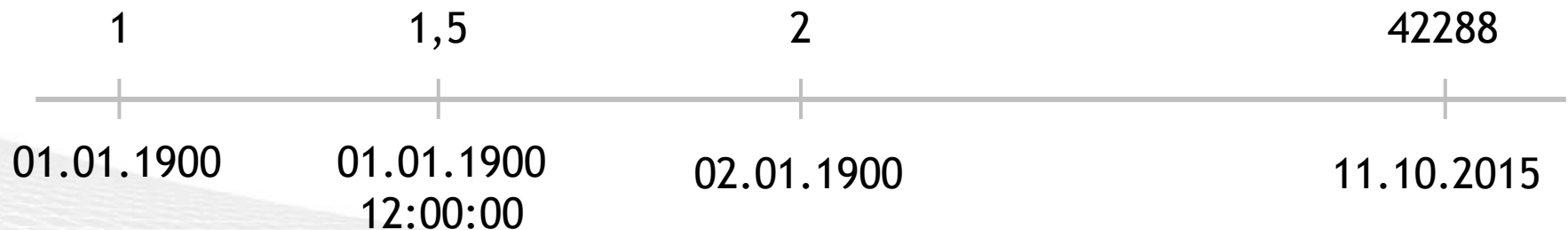
# Excel Funktionen

Kategorie Datum & Zeit



# Datum und Zeit in Excel

- ▶ Datums- und Zeitwerte werden intern als Zahlen behandelt
  - Ermöglicht das Rechnen mit Datums- und Zeitwerten
- ▶ Stichtag: 31.12.1899
  - Datumswerte danach werden als Abstand in Tagen vom Stichtag interpretiert (01.01.1900 entspricht der Zahl 1)
  - Der 31.12.1899 und Datumswerte davor werden **nicht** als gültiges Datum gesehen
  - Zeitwerte werden als Werte hinter der Dezimalstelle interpretiert (01.01.1900 12:00:00 entspricht der Zahl 1,5)



# HEUTE- und JETZT-Funktion

- ▶ Auslesen des aktuellen Datums bzw. des aktuellen Datums und der aktuellen Uhrzeit

Anweisung	Ergebnis
=Heute()	Aktuelles Datum
=Jetzt()	Aktuelles Datum und aktuelle Uhrzeit

- ▶ Um den Datums- bzw. Zeitwert als Zahl anzuzeigen, weisen Sie der Zelle das Standard- oder ein Zahlenformat zu

# JAHR-, MONAT- und TAG-Funktion

- ▶ Extrahieren eines Datumsteiles

Anweisung	Ergebnis
=Jahr("05.10.2015")	2015
=Monat("05.10.2015")	10
=Tag("05.10.2015")	5

- ▶ Auf den Folien arbeite ich mit literalen Datumskonstanten (Beispiel: "05.10.2015"). Dies kann im Echtbetrieb zu Problemen führen. Verwenden Sie die Funktionen DATUM bzw. DATWERT um ein entsprechendes Datum zu generieren. Oder noch besser: Sie arbeiten mit Zellbezügen, die Ihre Datumswerte enthalten.

# DATWERT-Funktion

- ▶ Datum als literale Konstante
  - Je nach Kontext ist Excel in der Lage eine implizite Konvertierung vorzunehmen
  - =Jahr("01.01.2015") liefert die Zahl 2015, da Excel versucht, das Argument als Datum zu interpretieren
  - =Wenn("05.02.2015" < "10.01.2015"; 0; 1) liefert die Zahl 0, da Excel die beiden Datumsangaben als Text vergleicht. Die 0 kommt somit alphabetisch vor der 1.
- ▶ DATWERT-Funktion
  - Wandelt eine literale Datumskonstante in ein Datum (also eigentlich in eine Zahl) um
  - Etwaige Zeitkomponente wird ignoriert \*)
  - =WENN( DatWert("05.02.2015") < DatWert("10.01.2015"); 0; 1) liefert jetzt das korrekte Ergebnis 1

\*) Recherchieren Sie selbst die Funktion ZEITWERT.

# DATUM-Funktion

- ▶ Erzeugt ein gültiges Datum aus den Angaben für die Jahreszahl, den Monat und den Tag

Anweisung	Ergebnis
=Datum( 2010; 12; 24 )	24.12.2010
=Datum( 2010; 13; 24 )	24.01.2011
=Datum( 2010; -1; 24 )	24.11.2009
=Datum( 2010; 12; 32 )	01.01.2011
=Datum( 2010; 12; -1 )	29.11.2010
=Datum( Jahr( Heute() ); 12; 31)	Letzter Tag des aktuellen Jahres
=Datum( Jahr( Heute() ) - 1; 12; 31)	Letzter Tag des vorherigen Jahres

\*) Recherchieren Sie selbst die Funktion ZEIT.

# WOCHENTAG-Funktion

- ▶ Liefert die Zahl des Wochentages (1 - 7 bzw. 0 - 6)
- ▶ Zwei Argumente
  - Datumsangabe (als Datum oder literale Konstante)
  - Zahl, die steuert, bei welchem Wochentag die Zählung beginnen soll

Anweisung *)	Ergebnis	Bemerkung
=Wochentag( "05.10.2015"; 1 )	2	Sonntag 1 - Samstag 7
=Wochentag( "05.10.2015"; 2 )	1	Montag 1 - Sonntag 7
=Wochentag( "05.10.2015"; 3 )	0	Montag 0 - Sonntag 6
=Wochentag( DatWert("05.10.2015") ; 1 )	2	Sonntag 1 - Samstag 7 Explizite Konvertierung mit DATWERT-Funktion

\*) 05.10.2015 fällt auf einen Montag.

# MONATSENDE-Funktion

- ▶ Berechnet den Monatsletzten
- ▶ Zwei Argumente
  - Datumsangabe (als Datum oder literale Konstante)
  - Anzahl der Monate, die zur Datumsangabe hinzugezählt bzw. abgezogen werden

Anweisung	Ergebnis
=Monatsende( "05.10.2015"; 0 )	31.10.2015
=Monatsende ( "05.10.2015"; 2 )	31.12.2015
=Monatsende ( "05.10.2015"; -1 )	30.09.2015
=Monatsende ( "05.10.2015"; -1 ) + 1	01.10.2015
=Monatsende ( DatWert("05.10.2015") ; 0 )	31.10.2015
=Monatsende ( Heute(); 0 )	Monatsende des aktuellen Datums



# NETTOARBEITSTAGE-Funktion

- ▶ Anzahl der Tage zwischen zwei Datumsangaben, die nicht auf einen Samstag bzw. Sonntag fallen
- ▶ Drei Argumente
  - Startdatum
  - Enddatum
  - Optionale Angabe einer Liste von Feiertagen möglich (als Matrixkonstante oder als Zellbereich)

Anweisung	Ergebnis
=Nettoarbeitstage( "01.12.2015";"31.12.2015" )	23
=Nettoarbeitstage( "01.12.2015";"31.12.2015"; {"25.12.2015"; "26.12.2015"} )	22 (der 26.12.2015 ist ein Samstag)
=Nettoarbeitstage( "01.12.2015";"31.12.2015"; C1:C2)	22 (in C1:C2 stehen die beiden Feiertagsangaben)

# NETTOARBEITSTAGE.INTL-Funktion

- ▶ Zusätzliche Angabe, welche Wochentage als freie Tage gelten
- ▶ Vier Argumente
  - Startdatum
  - Enddatum
  - Wochenende
    - Zahlenkonstante, die die freien Wochentage repräsentiert (Beispiel 11 → Nur Sonntag)
    - Zeichenkette aus 7 Ziffern, wobei die erste Ziffer den Montag repräsentiert (Beispiel "0010011" → Mittwoch, Samstag und Sonntag sind freie Wochentage)
  - Optionale Angabe einer Liste von Feiertagen möglich (als Matrixkonstante oder als Zellbereich)

# NETTOARBEITSTAGE.INTL-Funktion

Anweisung	Ergebnis
=Nettoarbeitstage.Intl( "01.12.2015";"31.12.2015"; 1 )	23 (Samstag und Sonntag freier Tag)
=Nettoarbeitstage.Intl( "01.12.2015";"31.12.2015"; 11 )	27 (Sonntag freier Tag)
=Nettoarbeitstage.Intl( "01.12.2015";"31.12.2015"; "0010011" )	18 (Mittwoch, Samstag und Sonntag freier Tag)
=Nettoarbeitstage( "01.12.2015";"31.12.2015"; 1; { "25.12.2015"; "26.12.2015" } )	22 (der 26.12.2015 ist ein Samstag)
=Nettoarbeitstage( "01.12.2015";"31.12.2015"; 1; <b>C1:C2</b> )	22 (in C1:C2 stehen die beiden Feiertagsangaben)

# ARBEITSTAG-Funktion

- ▶ Datum, das eine bestimmte Anzahl von Arbeitstage von einem Ausgangsdatum entfernt ist.
  - Samstag und Sonntag gelten nicht als Arbeitstage
- ▶ Drei Argumente
  - Ausgangsdatum
  - Anzahl Arbeitstage
    - Positive Zahl → Verschiebung in die Zukunft
    - Negative Zahl → Verschiebung in die Vergangenheit
  - Optionale Angabe einer Liste von Feiertagen möglich (als Matrixkonstante oder als Zellbereich)

# ARBEITSTAG-Funktion

Anweisung	Ergebnis
=Arbeitsstag( "01.12.2015"; 10 )	15.12.2015
=Arbeitsstag( "01.12.2015"; -10 )	17.11.2015
=Arbeitsstag( "01.12.2015"; 30; {"25.12.2015"; "26.12.2015"; "01.01.2016"} )	14.01.2016
=Arbeitsstag ( "01.12.2015"; 30; <b>C1:C3</b> )	22 (in C1:C3 stehen die drei Feiertagsangaben)

# ARBEITSTAG.INTL-Funktion

- ▶ Zusätzliche Angabe, welche Wochentage als freie Tage gelten
- ▶ Vier Argumente
  - Ausgangsdatum
  - Anzahl Arbeitstage
    - Positive Zahl → Verschiebung in die Zukunft
    - Negative Zahl → Verschiebung in die Vergangenheit
  - Wochenende
    - Zahlenkonstante, die die freien Wochentage repräsentiert (Beispiel 11 → Nur Sonntag)
    - Zeichenkette aus 7 Ziffern, wobei die erste Ziffer den Montag repräsentiert (Beispiel "0010011" → Mittwoch, Samstag und Sonntag sind freie Wochentage)
  - Optionale Angabe einer Liste von Feiertagen möglich (als Matrixkonstante oder als Zellbereich)

# ARBEITSTAG.INTL-Funktion

Anweisung	Ergebnis
=Arbeitsstag.Intl( "01.12.2015"; 10; 1 )	15.12.2015 (Samstag und Sonntag freier Tag)
=Arbeitsstag.Intl( "01.12.2015"; 10; 11 )	12.12.2015 (Sonntag freier Tag)
=Arbeitsstag.Intl( "01.12.2015"; 30; "0010011" { "25.12.2015"; "26.12.2015"; "01.01.2016" } )	26.01.2016 (Mittwoch, Samstag und Sonntag freier Tag; Drei Feiertage)
=Arbeitsstag.Intl( "01.12.2015"; 30; "0010011"; C1:C3)	26.01.2016 (Mittwoch, Samstag und Sonntag freier Tag; Drei Feiertage in C1 - C3)