Account Receivables balance report for <<if [Accounts.Count() < 5]>> << foreach [in Accounts] >> <<[Account.Name]>><</foreach>><<else>><<[Accounts.Count()]>> accounts<</if>>

|  |  |
| --- | --- |
| Legal entity | <<[LegalEntity]>> |
| Period: | <<[PeriodStart] : “yyyy-MM-dd”>> to <<[PeriodEnd]: “yyyy-MM-dd”>> |
| Created: | <<[Created] : “yyyy-MM-dd” >> |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Account name - Account number* |  |  |  |  |  |
| Invoice number | Invoice date | Due date | Currency | Amount | Amount (Base) | Balance | Balance (Base) |
| *<<foreach [in Accounts]>>****<<[Account.Name]>>*** *-* ***<<[Account.AccountNumber]>>*** |  |  |  |
| <<foreach [in Invoices]>><<[InvoiceNumber]>> | <<[InvoiceDate.ToString("yyyy-MM-dd")]>> | <<[DueDate.ToString("yyyy-MM-dd")]>> | <<[TotalAmount.CurrencyCode]>> | <<[TotalAmount.Amount.ToString(“N2”, Culture.Sweden)]>> | <<[TotalAmount. BaseCurrencyAmount.ToString(“N2”, Culture.Sweden)]>> | <<[(TotalAmount.Amount – PaymentSettlements.Sum(s => s.Amount.Amount)).ToString(“N2”, Culture.Sweden)]>> | <<[(TotalAmount.BaseCurrencyAmount – PaymentSettlements.Sum(s => s.Amount. BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>><</foreach>> |
|  |  |  | **TOTAL** |  | <<[Invoices.Sum(i => i.TotalAmount.BaseCurrencyAmount).ToString(“N2”, Culture.Sweden)]>> |  | <<[Invoices.Sum(i => i.TotalAmount.BaseCurrencyAmount – i.PaymentSettlements.Sum(s => s.Amount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
|  |  |  |  |  |  |  | <</foreach>> |

Number of invoices: <<[ Accounts.Sum(a => a.Invoices.Count()) ]>>
Total basecurrency: <<[ Accounts.Sum(a => a.Invoices.Sum(i => i.TotalAmount.BaseCurrencyAmount - i.PaymentSettlements.Sum(s => s.Amount. BaseCurrencyAmount))).ToString(“N2”, Culture.Sweden) ]>>

**Aging report:**

|  |  |  |
| --- | --- | --- |
| **Aging Buckets (days past due)** | **Number of invoices** | **Past due amount (base)** |
| Not due yet | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue == null).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue == null).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| 1-15 days | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 1 && i.DaysPastDue <= 15).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 1 && i.DaysPastDue <= 15).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| 16-30 days | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 16 && i.DaysPastDue <= 30).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 16 && i.DaysPastDue <= 30).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| 31-60 days | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 31 && i.DaysPastDue <= 60).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 31 && i.DaysPastDue <= 60).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| 61-90 days | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 61 && i.DaysPastDue <= 90).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 61 && i.DaysPastDue <= 90).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| 91+ days | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 91).Count())]>> | <<[Accounts.Sum(a => a.Invoices.Where(i => i.DaysPastDue != null && i.DaysPastDue >= 91).Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>> |
| **Grand Total** | **<<[Accounts.Sum(a => a.Invoices. Count())]>>** | **<<[Accounts.Sum(a => a.Invoices.Sum(i => i.UnsettledAmount.BaseCurrencyAmount)).ToString(“N2”, Culture.Sweden)]>>** |