

Prueba de R/Python

Parte 1. Lectura y limpieza de datos

1. Los datos a usar durante toda la prueba se encuentran en el archivo *Datos.zip*. El primer paso consiste en leer los archivos *Creditos* y *Calendario* en R/Python.

Parte 2. Manipulación de datos

2. Los archivos pertenecen a datos de una cartera de crédito. A partir de esos datos, debe calcularse 2 de los siguientes tipos de amortización para cada crédito:

a. *Amortización Normal*

- i. Una vez leídos los datos, se debe calcular el pago que amortiza en su totalidad el crédito tomando en cuenta la tasa contractual y la tasa iva.
- ii. El siguiente paso es crear la tabla de amortización sobre la base de datos *Calendario.xlsx*, los campos que debe contener esta tabla son: Fecha Inicial, Fecha Final, Saldo Inicial, Pago, Intereses, Pago Capital, IVA, Saldo Insoluto/Saldo Final.
- iii. Posteriormente, se construirá una segunda tabla de amortización, en la que ahora el saldo inicial sea igual al Saldo Neto Financiado; para esta tabla, se debe calcular la TIR (tasa efectiva) que descuenta exactamente los flujos de efectivo previstos (el pago calculado anteriormente) y que este sea igual al Saldo Neto Financiado. Esta segunda tabla de amortización deberá construirse sobre las fechas de la primera tabla y el pago calculado para la primera tabla, y la cual debe contener los siguientes rubros: Fecha Inicial, Fecha Final, Saldo Neto Financiado Inicial, Pago, Intereses, Pago Capital, Ingresos Amortizados, Gastos Amortizados, Saldo Neto Financiado Insoluto/Saldo Neto Financiado Final.

b. *Amortización AportaciónCapitalPago*

Se efectúa un procedimiento análogo que para la *Amortización Normal*, pero considerando los cambios señalados para este tipo de amortización en la hoja *Comparativa* del archivo *Ejemplos.xlsx* (deben generalizarse los cambios; es decir, con cantidades -aumentos o disminuciones- arbitrarias para cada crédito).

c. *Amortización AportaciónCapitalCalendario*

Se efectúa un procedimiento análogo que para la *Amortización Normal*, pero considerando los cambios señalados para este tipo de amortización en la hoja *Comparativa* del archivo *Ejemplos.xlsx* (deben generalizarse los cambios; es decir, con cantidades arbitrarias -aumentos o disminuciones- para cada crédito).

Parte 3. Visualización de datos

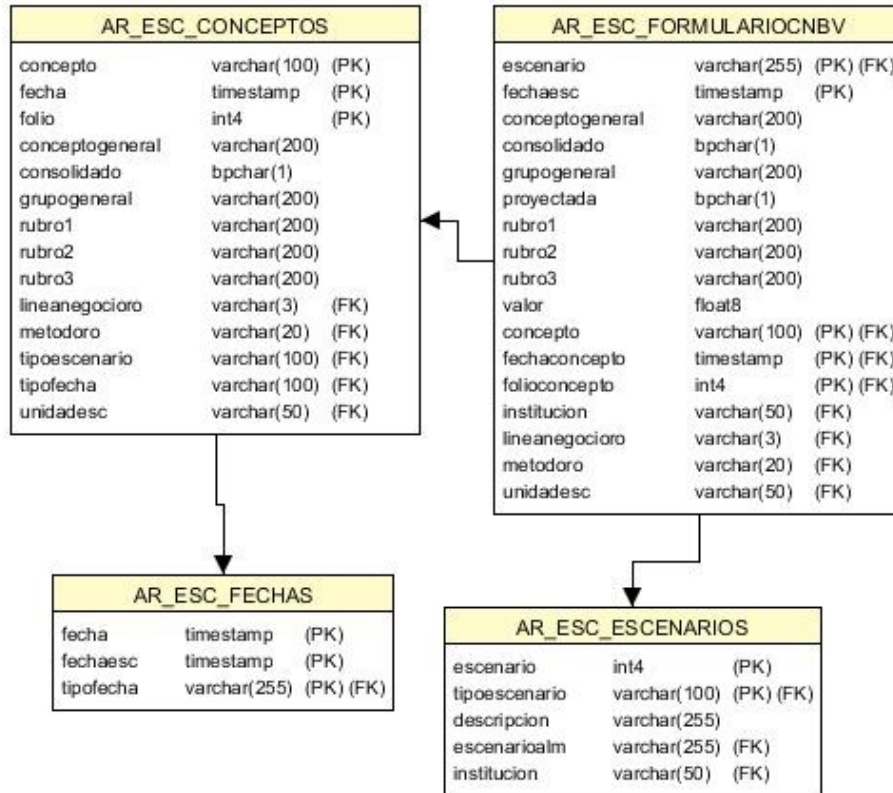
6. A partir de los datos calculados, imprima en *R Markdown* un ejemplo de las cuatro tablas obtenidas.

Hints/Ayudas

- El IVA corresponde al 16 % y se calcula solamente sobre los intereses.
- El Saldo Insoluto Final para ambas tablas de cada crédito debe ser igual o casi 0.
- Para los tres tipos de amortización: *Normal*, *AportaciónCapitalPago* y *AportaciónCapital-Calendario*, se muestra un ejemplo en el archivo **Ejemplos.xlsx** (lo que hay que realizar es replicar estos comportamientos para cada uno de los créditos de la cartera).

Prueba de SQL

Se tiene la siguiente estructura de datos. La tabla `ar_esc_formulariocnbv` se genera a partir de las tablas `ar_esc_conceptos`, `ar_esc_fechas`, `ar_esc_escenarios`.



1. Realiza una consulta que dé como resultado la tabla `ar_esc_formulariocnbv` uniendo las tablas antes mencionadas, y que filtre tal que la fecha sea igual a 30/09/2021, el concepto empiece por “Supuestos CNBV.1” y el `escenarioalm` se encuentre entre 2 y 4.
2. Supón que el query que acabas de realizar te devuelve 200 millones de datos, lo lees y almacenas en un `data.frame` en R/Python, y se requiere que te conectes a la base de datos de SQL, e insertes los datos del `data.frame` en una tabla de respaldo (`ar_esc_formulariocnbvrespaldo`) la cual cuenta ya con 800 millones de registros. ¿Cómo los insertarías a la tabla?