\*&---------------------------------------------------------------------\*  
\*& Report  Z\_OBJETO  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
  
REPORT  z\_objeto.  
  
  
\*----------------------------------------------------------------------\*  
\*       CLASS mountainbike DEFINITION  
\*----------------------------------------------------------------------\*  
\* Define Classe dinamicamente.  
\*----------------------------------------------------------------------\*  
CLASS mountainbike DEFINITION.  
  
  PUBLIC SECTION.  
  
    DATA modelo TYPE string .  
    DATA quadro TYPE string .  
    DATA freios TYPE string .  
    DATA rodas  TYPE string .  
  
    METHODS constructor IMPORTING p\_modelo TYPE string  
                                  p\_quadro TYPE string  
                                  p\_freios TYPE string  
                                  p\_rodas  TYPE string.  
  
    METHODS bicicleta .  
  
ENDCLASS.                    "MOUNTAINBIKE DEFINITION  
  
  
\*----------------------------------------------------------------------\*  
\*       CLASS MOUNTAINBIKE IMPLEMENTATION  
\*----------------------------------------------------------------------\*  
\*  
\*----------------------------------------------------------------------\*  
CLASS mountainbike IMPLEMENTATION.  
\*  
\*  
  METHOD bicicleta.  
  
    WRITE: / ' Bicicleta:', modelo." Minha implementação.  
    WRITE: / ' Quadro:'   , quadro.  
    WRITE: / ' Freios:'   , freios.  
    WRITE: / ' Rodas:'    , rodas.  
  
  ENDMETHOD.                    "BICICLETA  
\*  
\*  
  METHOD constructor.  
  
    me->modelo = p\_modelo.  
    me->quadro = p\_quadro.  
    me->freios = p\_freios.  
    me->rodas  = p\_rodas.  
  
  ENDMETHOD.                    "CONSTRUCTOR  
\*  
\*  
ENDCLASS.                    "MOUNTAINBIKE IMPLEMENTATION  
  
  
  
\*----------------------------------------------------------------------\*  
\*       START-OF-SELECTION  
\*----------------------------------------------------------------------\*  
\* Cria objetos Bicicleta - definidos na CLASS BUILDER.  
\*----------------------------------------------------------------------\*  
  
  
START-OF-SELECTION.  
  
  DATA bike1 TYPE REF TO zbicicleta.  
  DATA bike2 TYPE REF TO zbicicleta.  
  
  CREATE OBJECT bike1.  
  CREATE OBJECT bike2.  
  
  WRITE: / .  
  WRITE: / '\*\_\_\_\_\_\_\_Bicicleta #1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*'.  
  WRITE: / .  
  
  
\*\*\* Alimentando as Variáveis  
  
\* Objeto Bicicleta #1  
  MOVE 'BMX'       TO bike1->modelo.  
  MOVE 'Ferro'     TO bike1->quadros.  
  MOVE 'Pastilhas' TO bike1->freios.  
  MOVE 'Ferro'     TO bike1->rodas.  
  
\* Objeto Bicicleta #2  
  MOVE 'Caloi'     TO bike2->modelo.  
  MOVE 'Aluminio'  TO bike2->quadros.  
  MOVE 'Pastilhas' TO bike2->freios.  
  MOVE 'Aluminio'  TO bike2->rodas.  
  
  CALL METHOD bike1->bicicleta.”dados da bike1 carregadas então, chama o metodo bicicleta  
  
  WRITE: / .  
  WRITE: / '\*\_\_\_\_\_\_\_Bicicleta #2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*'.  
  WRITE: / .  
  
  CALL METHOD bike2->bicicleta.” Dados da bike2 carregadas então, chama o meotod bicicleta

\* Objeto criado usando o  metodo Construtor  
  WRITE: / .  
  WRITE: / '\*\_\_\_\_\_\_\_Bicicleta #3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*'.  
  WRITE: / .  
  
  DATA bike3 TYPE REF TO mountainbike.  
  
  CREATE OBJECT bike3  
    EXPORTING  
      p\_modelo = 'ZX-Track'  
      p\_quadro = 'Fibra de Carbono'  
      p\_freios = 'Disco'  
      p\_rodas  = 'Aluminio'.  
  
  CALL METHOD bike3->bicicleta.

Execução poderia ser feita toda assim para os 3 biks

