Very Simple ASIS 95 Example - 1

```
with Text_IO; use Text_IO;
with Asis.Implementation, Asis.Ada_Environments, Asis.Compilation_Units;
with Asis.Elements, Asis.Declarations, Asis_Element_Kinds;
Procedure Asis_Application_Example is
   My_Context       : Asis.Context;
   My_Unit          : Asis.Compilation_Unit;
   Unit_Name        : String( 1 .. 1024 );
   Unit_Name_Length : Natural;
   procedure Report_Declarations( Unit : in Asis.Compilation_Unit ) is ...
begin -- Asis_Application_Example
   Asis.Implementation.Initialize;
   Asis.Ada_Environments.Associate( My_Context, "my_library_search_path_name" );
   Asis.Ada_Environments.Open( My_Context );

   Put_Line( "Type the name of an Ada package specification" );
   Get_Line( Unit_Name, Unit_Name_Length );

   My_Unit := Asis.Compilation_Units.Library_Unit_Declaration
                 ( Unit_Name( 1 .. Unit_Name_Length), My_Context );
   if Asis.Compilation_Units.Is_Nil( My_Unit ) then
      Put_Line( "Context does not contain the requested unit" );
   else
      Put_Line( "Context contains the requested unit" );
      Report_Declarations( My_Unit );
   end if;

   Asis.Ada_Environments.Close( My_Context );
   Asis.Ada_Environments.Dissociate( My_Context );
   Asis.Implementation.Finalize;
end Asis_Application_Example;
```

Very Simple ASIS 95 Example - 2

```
procedure Report_Declarations( Unit : in Asis.Compilation_Unit ) is

   My_Control   : Asis.Elements.Traverse_Control := Asis.Elements.Continue;
   Unit_Element : Asis.Element;
   Count        : Natural                        := 0;

   procedure Process_Element
      ( Elem    : in     Asis.Element;
        Control : in out Asis.Elements.Traverse_Control;
        Counter : in out Natural ) is ...

   procedure No_Op
      ( Elem    : in     Asis.Element;
        Control : in out Asis.Elements.Traverse_Control;
        State   : in out Natural ) is
   begin
      null;
   end No_Op;

   procedure Find_and_Print_Declarations is new
      Asis.Elements.Traverse_Element( Natural, Process_Element, No_Op );

begin -- Report_Declarations
   Unit_Element := Asis.Elements.Unit_Declaration( Unit );
   Find_and_Print_Declarations( Unit_Element, My_Control, Count );
   Put_Line( “Found” & Integer’Image( Count ) & “ declarations.” );
end Report_Declarations;
```

Very Simple ASIS 95 Example - 3

```
procedure Process_Element( Elem    : in     Asis.Element;
                           Control : in out Asis.Elements.Traverse_Control;
                           Counter : in out Natural )
is
   Decl_Kind : constant Asis_Element_Kinds.Declaration_Kinds :=
                           Asis.Elements.Declaration_Kind( Elem );
begin -- Process_Element
   case Decl_Kind is
      when Asis_Element_Kinds.Not_A_Declaration =>
         null;

      when others =>
         declare
            Name_List : constant Asis.Defining_Name_List :=
                                    Asis.Declarations.Names( Elem );
         begin
            for I in Name_List'Range loop
               Put( "The View " );
               Put( Asis.Declarations.Defining_Name_Image( Name_List( I ) ) );
               Put( " (is kind) " );
               Put( Asis_Element_Kinds.Declaration_Kinds’Image( Decl_Kind ) );
               New_Line;
               Counter := Counter + 1;
            end loop;
         end;

   end case;
   Control := Asis.Elements.Continue;   -- Actually not necessary in this example
end Process_Element;
```

Very Simple ASIS 95 Example - 4

```
For =>

package Asis_Test is
   type T is ( A, B, C);
   S : Integer := T'Base'Size;
end Asis_Test;
```

```
Result =>

Type the name of an Ada package specification
asis_test
Context contains the requested unit
The View ASIS_TEST (is kind) A_PACKAGE_DECLARATION
The View T (is kind) AN_ORDINARY_TYPE_DECLARATION
The View A (is kind) AN_ENUMERATION_LITERAL_SPECIFICATION
The View B (is kind) AN_ENUMERATION_LITERAL_SPECIFICATION
The View C (is kind) AN_ENUMERATION_LITERAL_SPECIFICATION
The View S (is kind) A_VARIABLE_DECLARATION
Found 6 declarations.
```