Public exam - GeneXus 17 Junior Developer

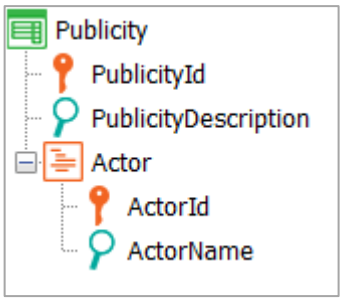
Reality: Advertising Agency.
About multiple choice questions:

- There is only one correct option.
- In this exam, NO points are deducted for incorrect answers.

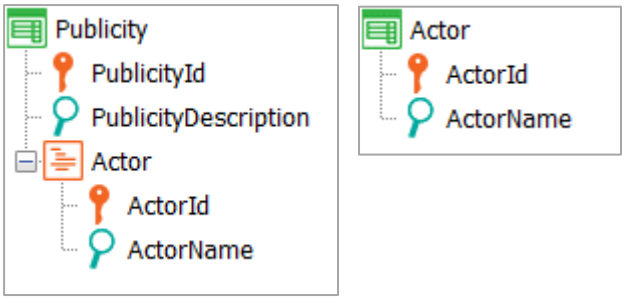
1) A GeneXus application is needed for an Advertising Agency.

Knowing that several actors (Actor) work in an advertisement (Publicity) and that an actor can appear in several advertisements, determine the transaction design that you consider correct to model the described reality.

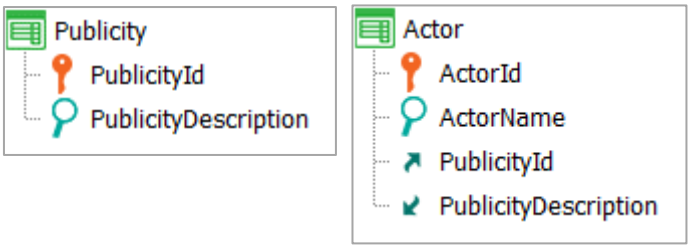
1.1 –



1.2 –



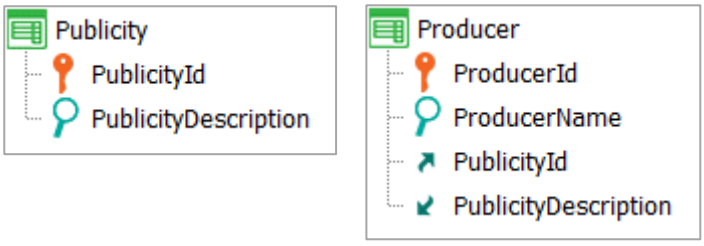
1.3 –



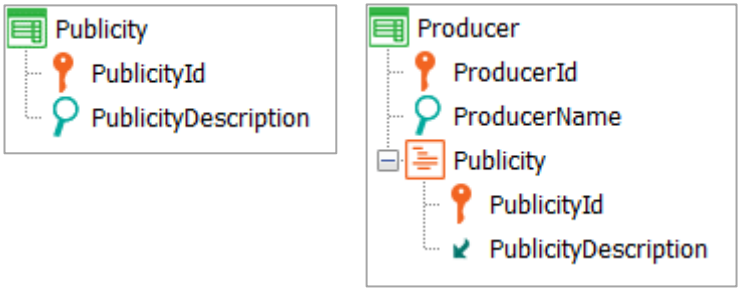
1.4 – None of the above options is correct.

2) Knowing that every advertisement (Publicity) has a producer (Producer) in charge, and that every producer is in charge of several advertisements, select the transaction design you consider correct to adequately model the reality described.

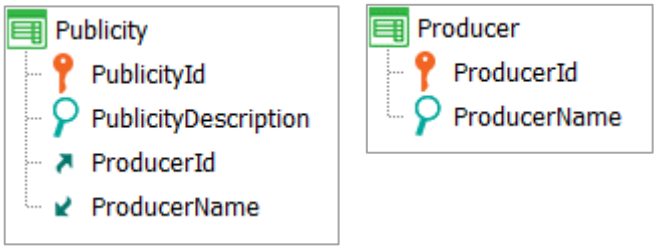
2.1 –



2.2 –



2.3 –

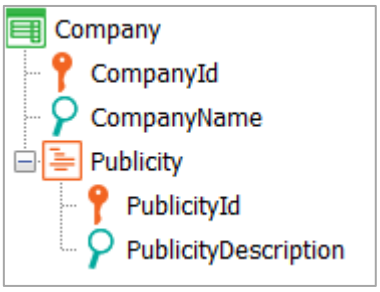


2.4 – None of the above options is correct.

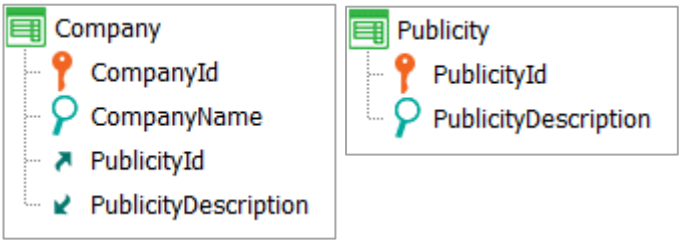
3) A client company (Company) purchases many advertisements (Publicity), which belong exclusively to that company. Determine the transaction design you consider correct to model the reality described.

3.1 –

3.2 –

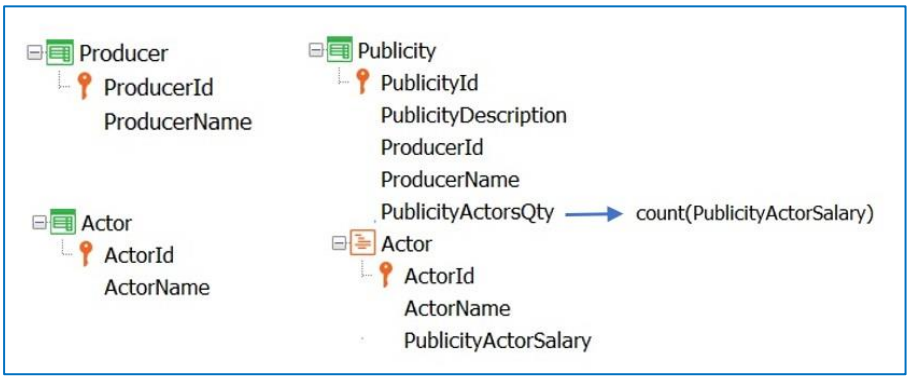


3.3 –



3.4 – None of the above options is correct.

4) Based on the transaction design displayed, determine the structure of the tables that will be created by GeneXus:



4.1 –

ACTOR	**PRODUCER**	**PUBLICITY**	**PUBLICITYACTOR**
ActorId*	ProducerId*	PublicityId*	PublicityId*
ActorName	ProducerName	PublicityDescription	ActorId*
		ProducerId	PublicityActorSalary

4.2 –

ACTOR	**PRODUCER**	**PUBLICITY**	**PUBLICITYACTOR**
ActorId*	ProducerId*	PublicityId*	PublicityId*
ActorName	ProducerName	PublicityDescription	ActorId*
		ProducerId	PublicityActorSalary
		PublicityActorsQty	

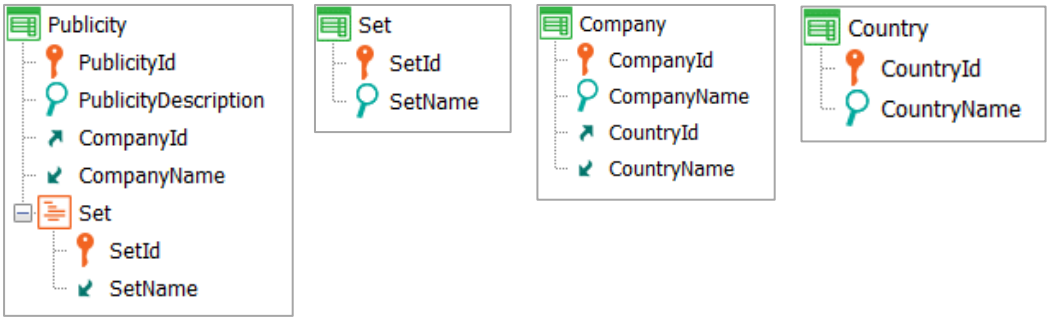
4.3 –

ACTOR	**PRODUCER**	**PUBLICITY**	**PUBLICITYACTOR**
ActorId*	ProducerId*	PublicityId*	PublicityId*
ActorName	ProducerName	PublicityDescription	ActorId*
		ProducerId	

4.4 –

ACTOR	**PRODUCER**	**PUBLICITYACTOR**
ActorId*	ProducerId*	PublicityId*
ActorName	ProducerName	PublicityDescription
		ProducerId
		ActorId*
		PublicityActorSalary

5) Look at the transaction design displayed and determine the extended table of the PUBLICITY base table.



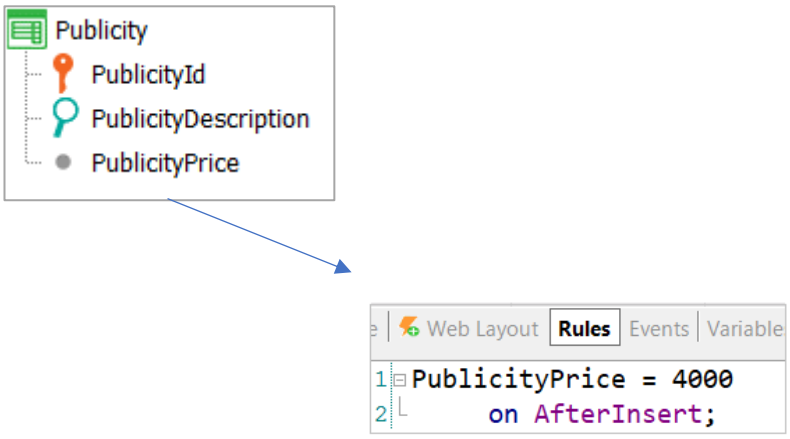
5.1 - PUBLICITY, SET

5.2 - PUBLICITY, COMPANY, SET

5.3 - PUBLICITY, COMPANY, COUNTRY

5.4 - PUBLICITY, COMPANY, COUNTRY, SET

6) Consider the transaction design and the declared rule that are displayed.



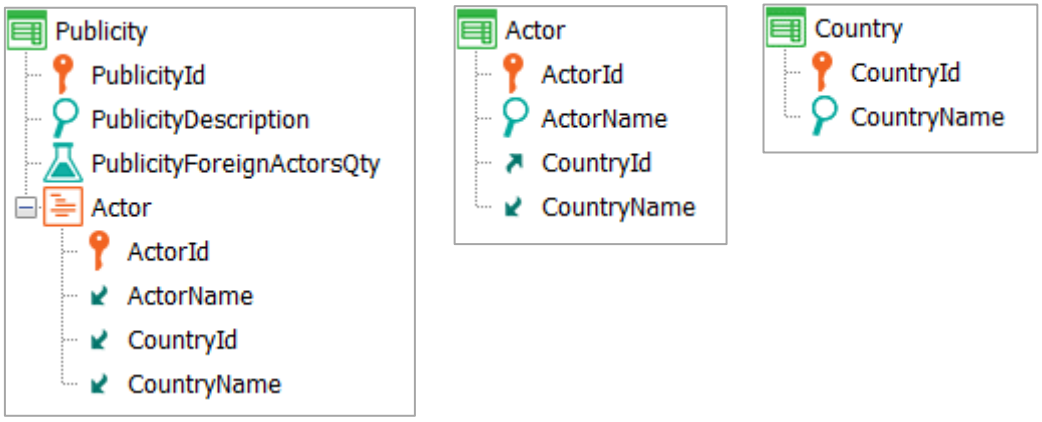
6.1 – The declared rule corresponds to a functionally correct definition.

6.2 – The declared rule doesn't correspond to a functionally correct definition, since it's not possible to condition a rule to be executed only if the transaction is accessed in Insert mode.

6.3 – The declared rule doesn't correspond to a functionally correct definition, because in the triggering event *on AfterInsert* the data was already entered in the Publicity table, so it is too late to attribute a value to an attribute.

6.4 – None of the above options is correct.

7) Consider the transaction design shown below.
We are asked to always know the number of foreign (CountryId <> 1) actors (Actor) who appear in each advertisement.
Determine the definition you consider correct for the formula attribute PublicityForeignActorsQty.



7.1 –

Count(ActorId, CountryId <> 1)

7.2 –

Count(ActorId) if CountryId <> 1

7.3 –

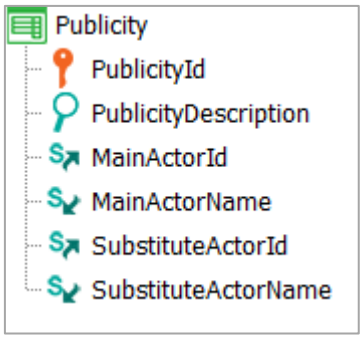
```
If CountryId <> 1
   Count(ActorId)
Endif
```

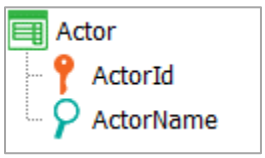
7.4 – None of the above options is correct.

8) Consider the transaction design displayed.

For each advertisement, it is necessary to register a primary actor and an alternate actor, in case the primary actor is unable to attend the recording.

Determine if the following implementation is correct (true) or not (false).





Subtype	Description	Supertype
MainActor		
MainActorId	Main Actor Id	ActorId
MainActorName	Main Actor Name	ActorName

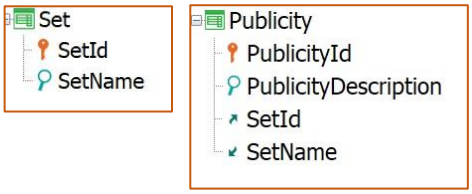
Subtype	Description	Supertype
SubstituteActor		
SubstituteActorId	Substitute Actor Id	ActorId
SubstituteActorName	Substitute Actor Name	ActorName

☐ True ☐ False

9) Consider the transaction design displayed.

Using the form of the Set transaction, the record with value SetId = 4 is to be deleted.

Determine what you consider correct:



9.1 – GeneXus deletes the record with value SetId=4 from the SET table without performing any controls.

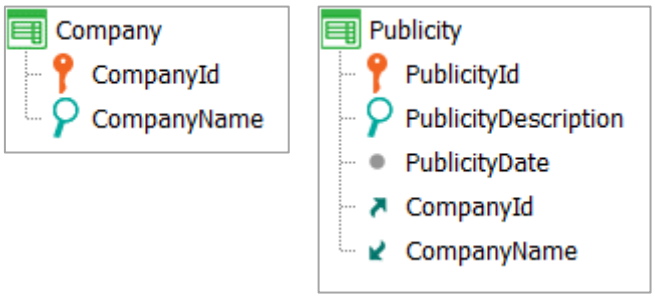
9.2 – GeneXus deletes the record with value SetId=4 from the SET table and also automatically deletes all the records of the PUBLICITY table that have this value as a foreign key SetId.

9.3 – GeneXus deletes the record with value SetId=4 from the SET table and then checks if in the PUBLICITY table there are still records with this value as a foreign key SetId. If there are any, it issues a warning message.

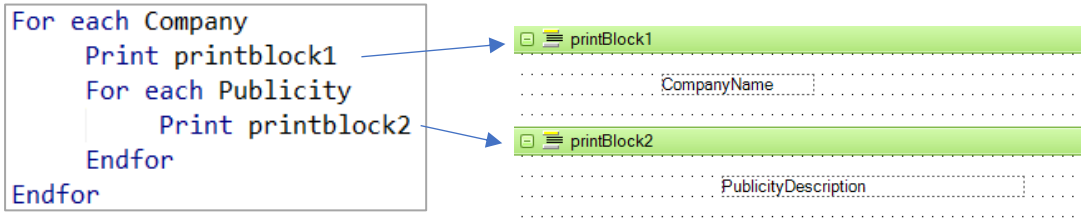
9.4 – GeneXus checks if the PUBLICITY table contains records with value 4 in its SetId foreign key. If there are any, it displays a message and doesn't perform any action.

10) Consider the transaction design displayed.

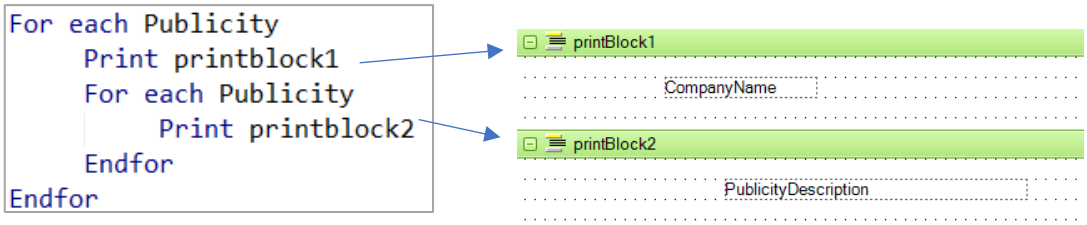
A list is requested that prints all the companies (Company), and for each one of them lists all the advertisements (Publicity) that they have purchased. All the companies should be listed, regardless of whether they have registered advertisements or not.



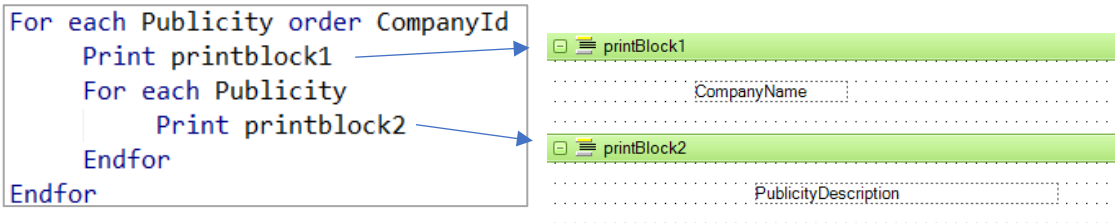
10.1 –



10.2 –



10.3 –

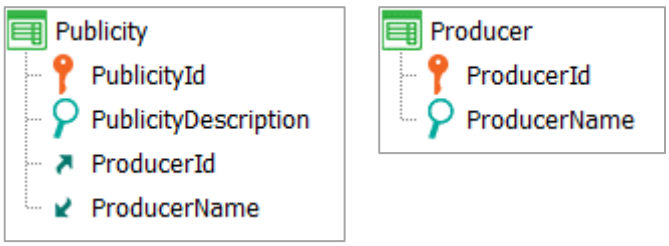


10.4 – None of the above options is correct.

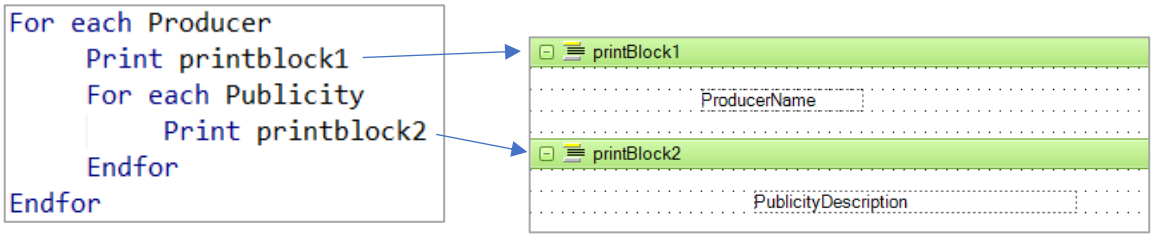
11) A list needs to be defined that shows all the advertisements (Publicity) grouped by their producer in charge (Producer).

Producers who do not have registered advertisements should not be printed.

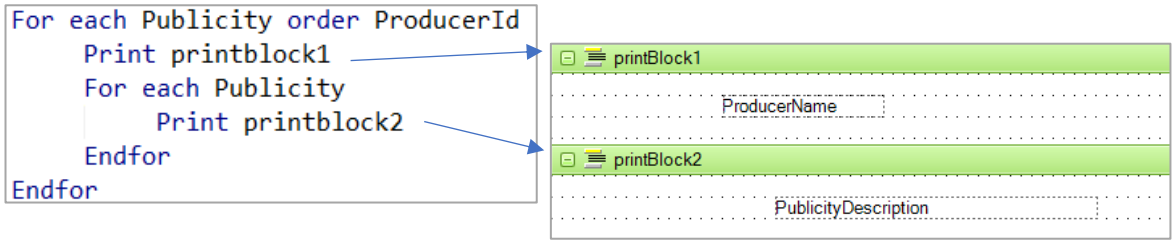
Determine the implementation option you consider correct to meet the requirement described.



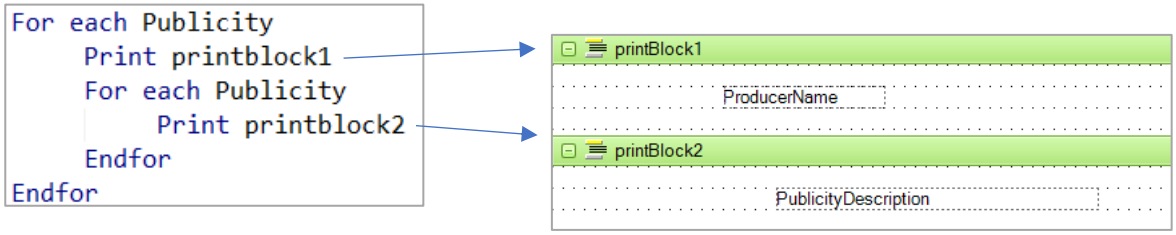
11.1 –



11.2 –

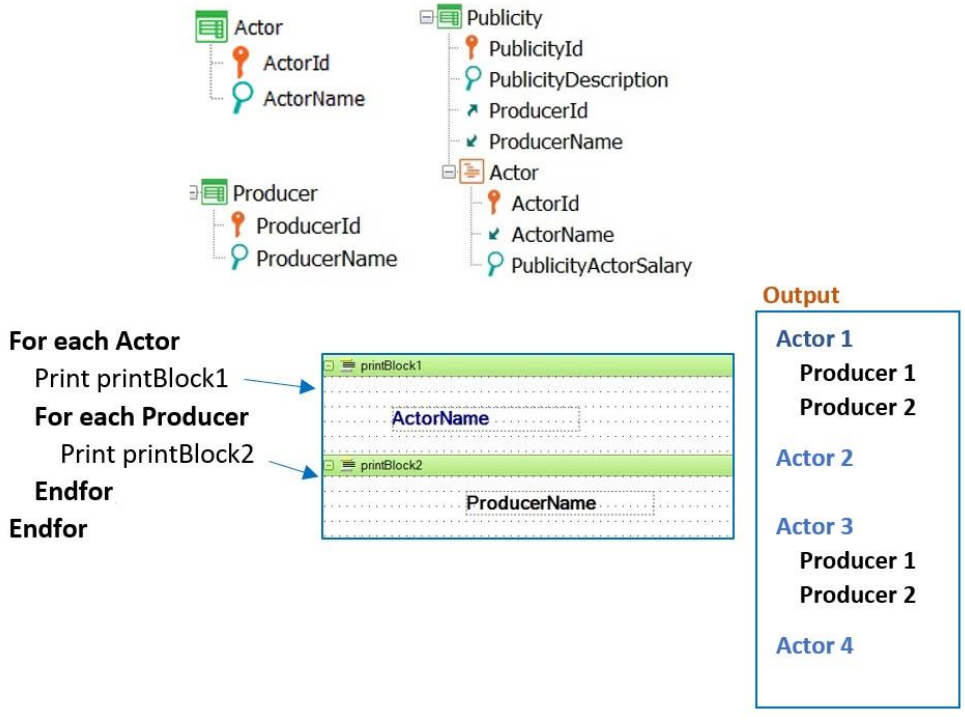


11.3 –



11.4 – None of the above options is correct.

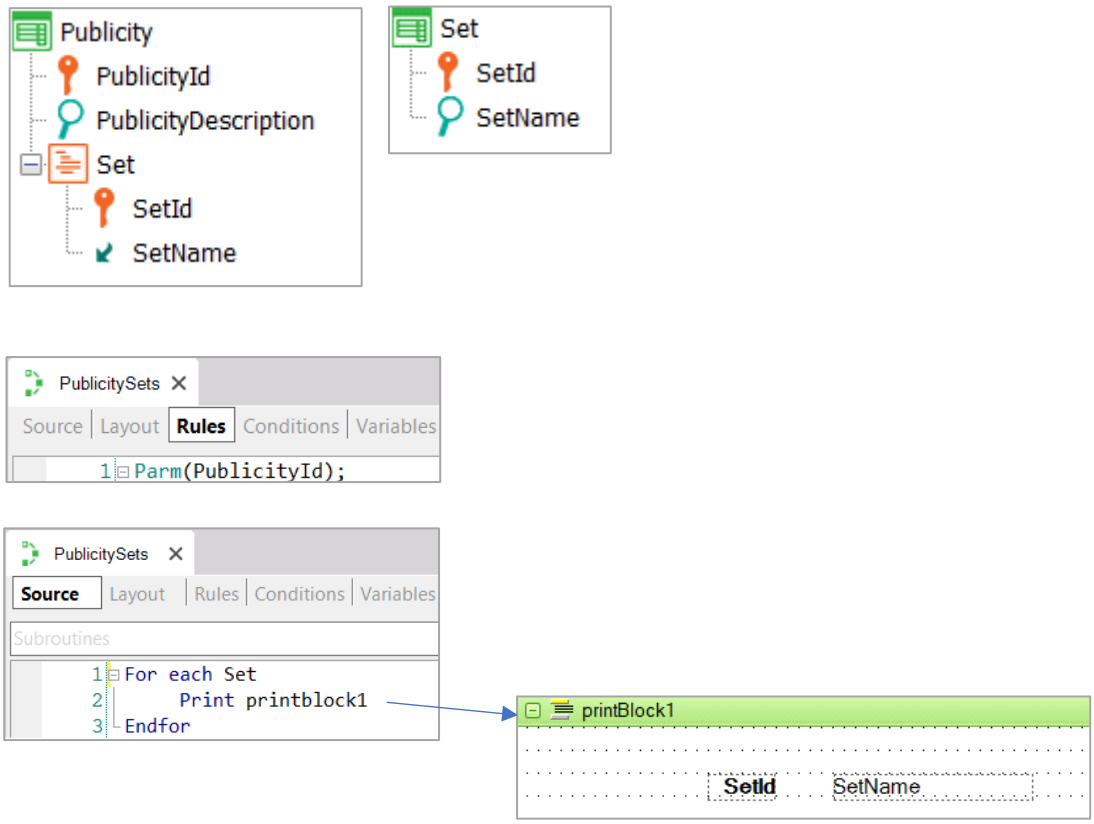
12) Consider the transaction design and implementation displayed. Determine if for the indicated source, the output is as shown.



☐ True ☐ False

13) Consider the transaction design displayed. It is necessary to receive the identifier of an advertisement (PublicityId) and list its film sets.

Determine what you consider correct.



13.1 **–** The implementation doesn't meet the requirement, because it will show all the existing sets, and not only those that belong to the advertisement received as a parameter. To be correct, the Base Transaction of the For Each must be Publicity.

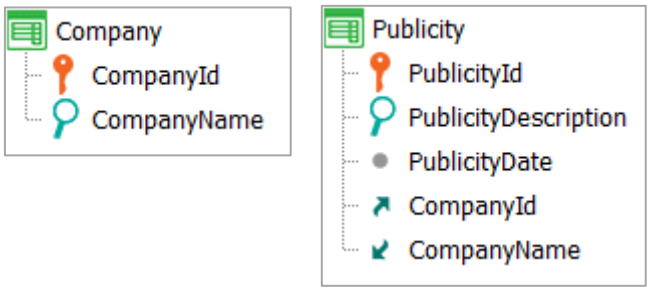
13.2 **–** The implementation doesn't meet the requirement, because it will show all the existing sets, and not only those that belong to the advertisement received as a parameter. To be correct, the Base Transaction of the For Each must be Publicity.Set.

13.3 **–** For GeneXus to automatically perform the filter by the received parameter, the Parm rule attribute must also be present in the printBlock.

13.4 **–** None of the above options is correct.

14) Consider the transaction design displayed.

A list has to be created of all the companies (Company) that have more than 5 advertisements (Publicity) registered.
Determine the implementation option you consider correct to meet the requirement described.

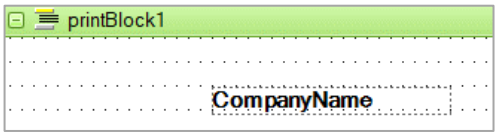


14.1 –

```
For each Company
     where count(PublicityDescription) > 5
     Print printBlock1
Endfor
```

14.2 –

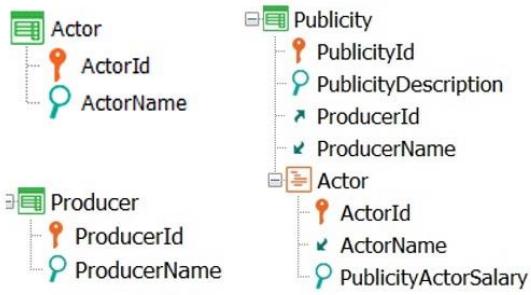
```
For each Publicity
     where count(PublicityDescription) > 5
     For each Company
          Print printBlock1
     Endfor
Endfor
```



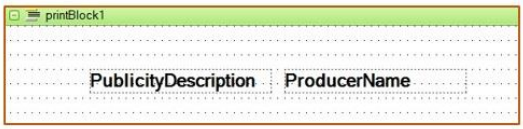
14.3 – It is not possible to declare inline formulas in a Where clause.

14.4 – None of the above options is correct.

15) Based on the transaction design and source displayed, determine the base table of the For Each:



For each Publicity
Where count(PublicityActorSalary) < 5
Print printBlock1
Endfor



15.1 – The base table of the For Each is PUBLICITYACTOR.

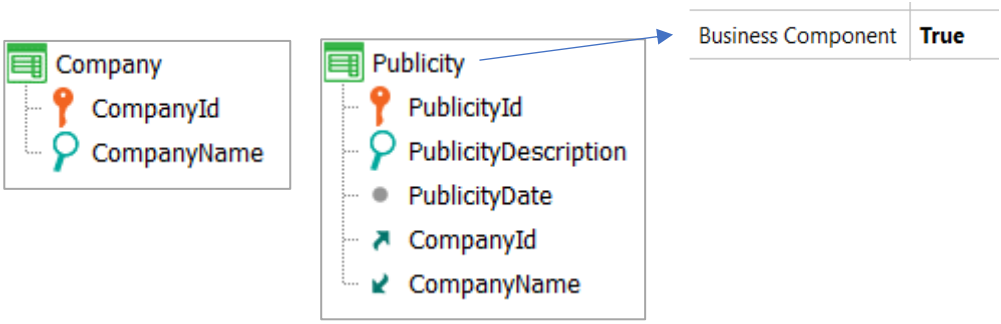
15.2 – The base table of the For Each is PRODUCER.

15.3 – The base table of the For Each is PUBLICITY.

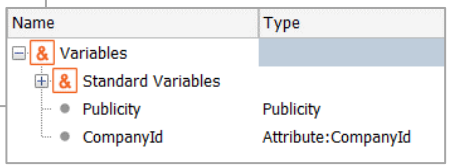
15.4 – The source is poorly defined since there is no extended table containing all the attributes declared in the For Each.

16) Consider the transaction design displayed.

All advertisements registered for a certain company (Company) received by parameter are requested to be deleted.
Determine whether the proposed implementation option is correct (true) or not (false) to solve the described requirement.



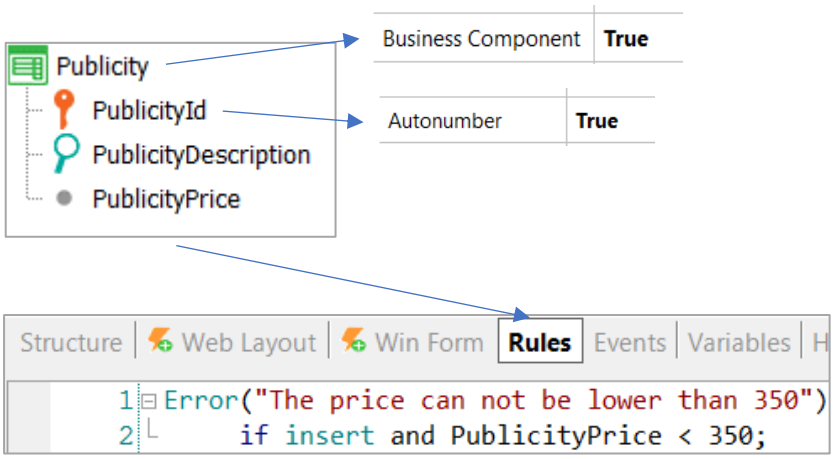

```
For each Publicity
     where CompanyId = &CompanyId
     &Publicity.Load(PublicityId)
     &Publicity.Delete()
Endfor
Commit
```



☐ True ☐ False

17) Consider the transaction design and source shown below.

Indicate whether the following statement is true or false: “The code displayed (which has been defined in an event or object source) will insert the new advertisement.”

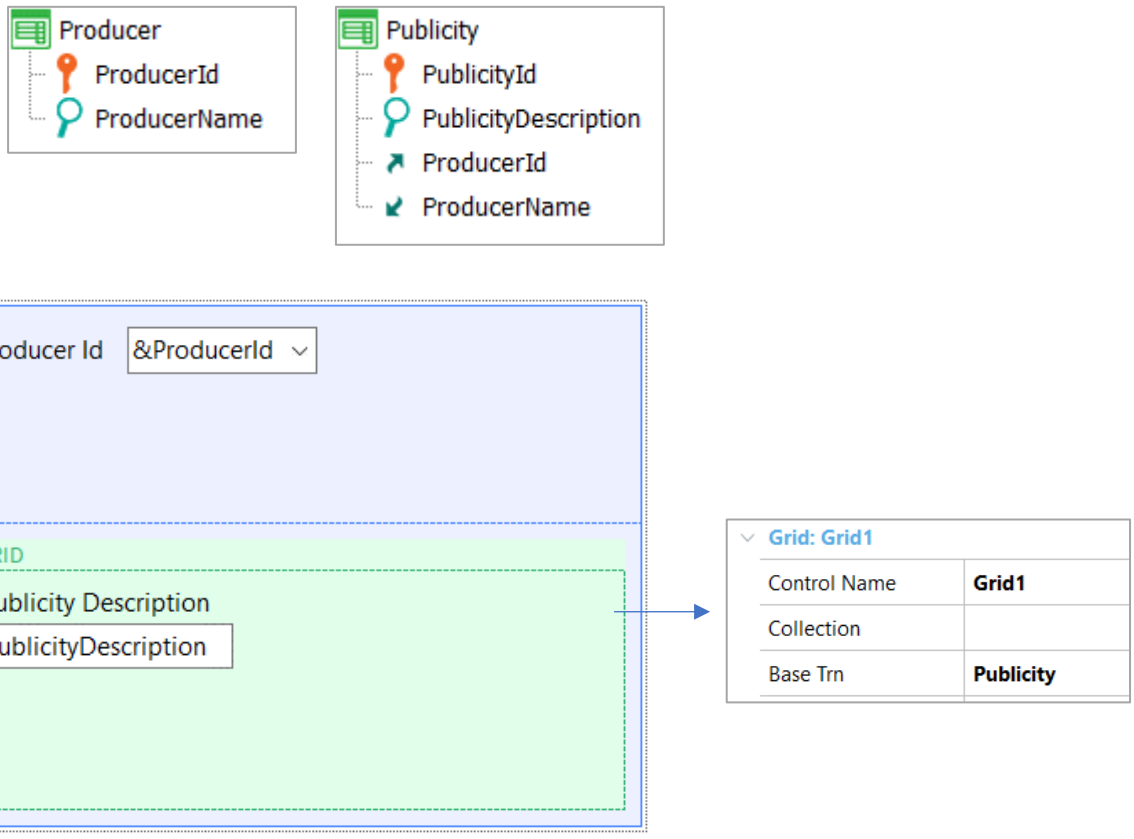



```
&Publicity.PublicityDescription = "Star publicity"
&Publicity.PublicityPrice = 200
&Publicity.Insert()
Commit
```

☐ True ☐ False

18) Consider the transaction design and the Web Panel displayed.

It is requested that the Web Panel displays the advertisements (Publicity) created by a producer (Producer) that the user has previously chosen from the Dynamic combo box offered.
Suppose that the dynamic combo box has been correctly created.
Select the option that you consider correct:

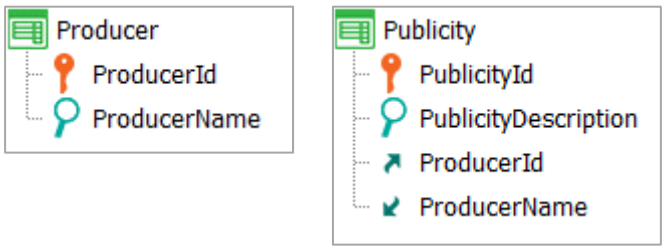


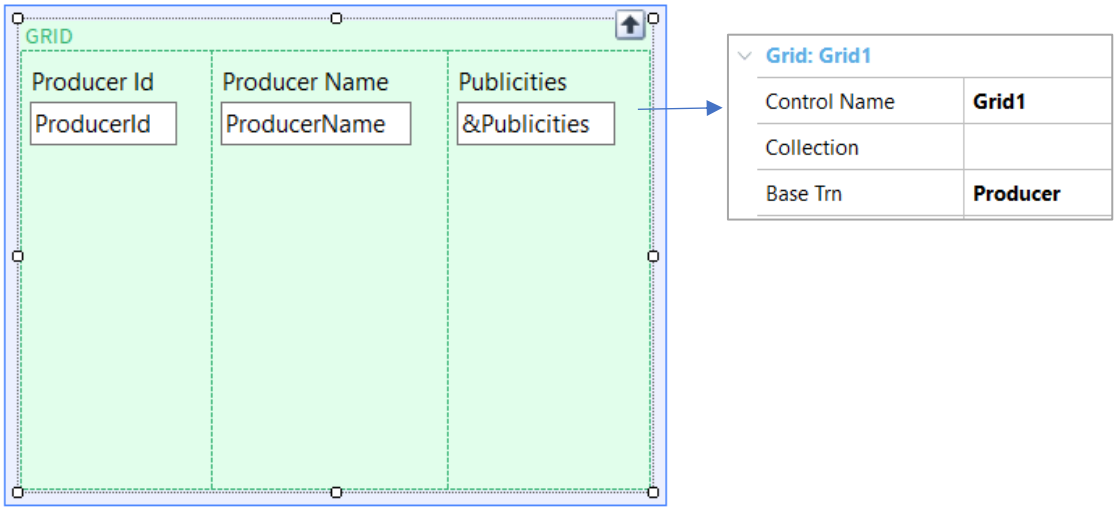
18.1 – It is not necessary to define anything else in the Web Panel displayed. GeneXus will automatically apply a filter by equality between the value of the variable and the attribute of the same name. Therefore, the grid will load the advertisements of the producer selected by the user.
18.2 – The following rule must be added: ProducerId=&ProducerId;

18.3 – The following condition must be added at the grid level:
ProducerId=&ProducerId;

18.4 – None of the above options is correct.

19) It is necessary to define a Web Panel that shows all the producers (Producer), each one with their corresponding number of registered advertisements (Publicity). Select the implementation option you consider correct to meet the requirement described.





19.1 –

```
Event Load
     &Publicities = Count(PublicityDescription)
Endevent
```

19.2 –

```
Event Load
     For each Producer
          &Publicities = Count(PublicityDescription)
     Endfor
Endevent
```

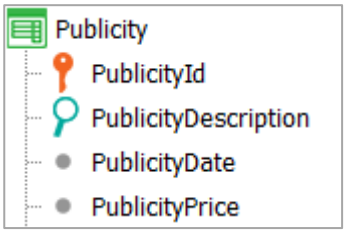
19.3 –

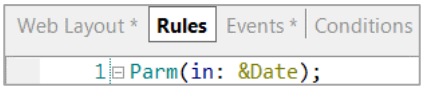
```
Event Load
     For each Producer
          &Publicities = Count(PublicityDescription)
          Load
     Endfor
Endevent
```

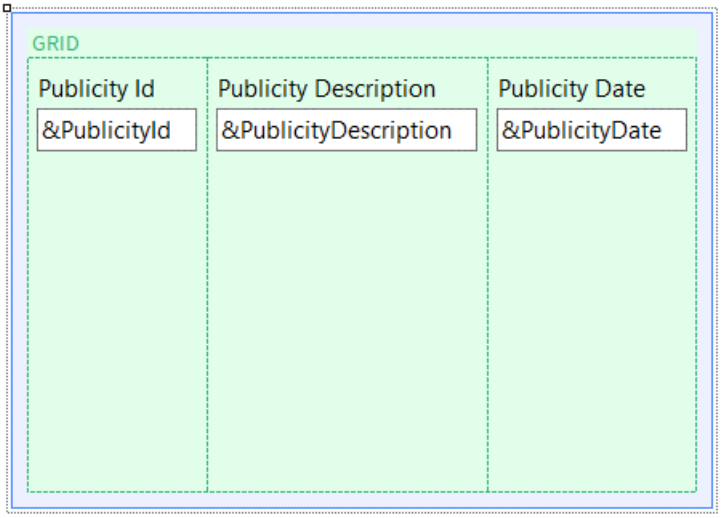
19.4 – None of the above options is correct.

20) Consider the transaction design displayed.

It is necessary to define a Web Panel that shows all the advertisements (Publicity) that have been registered as of a certain date received by parameter.
Select the implementation option you consider correct to meet the requirement described.







20.1 –

```
Event Load
     For each Publicity
          where PublicityDate >= &Date
          &PublicityId = PublicityId
          &PublicityDescription = PublicityDescription
          &PublicityDate = PublicityDate
     Endfor
Endevent
```

20.2 –

```
Event Load
     if PublicityDate >= &Date
          &PublicityId = PublicityId
          &PublicityDescription = PublicityDescription
          &PublicityDate = PublicityDate
          Load
     Endif
Endevent
```

20.3 –

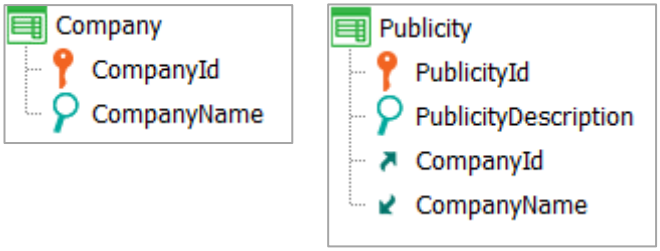
```
Event Load
     For each Publicity
          where PublicityDate >= &Date
          &PublicityId = PublicityId
          &PublicityDescription = PublicityDescription
          &PublicityDate = PublicityDate
          Load
     Endfor
Endevent
```

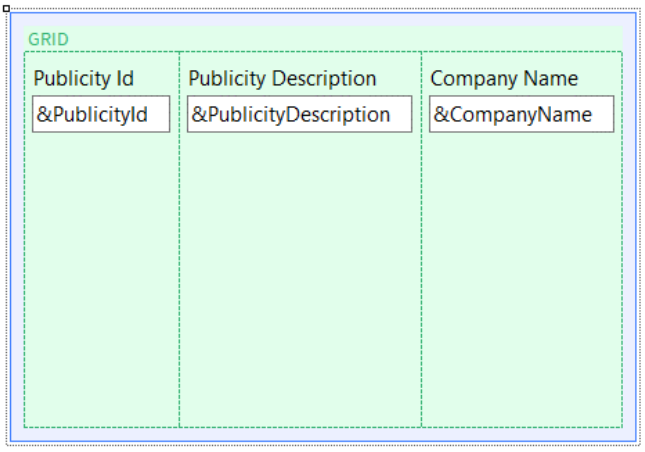
20.4 – None of the above options is correct.

21) Consider the transaction design and the Web Panel displayed.

Knowing that there are 100 companies (Company) and 1000 advertisements (Publicity) registered, determine the number of times that the Load event of the Web Panel will be executed.

Note: The object sections or properties that are not shown have not been changed from their default values.





21.1 – 1000 times

21.2 – 100 times

21.3 – Once

21.4 – Never

Answers

1) 2

2) 3

3) 2

4) 1

5) 3

6) 3

7) 1

8) True

9) 4

10) 1

11) 2

12) False

13) 2

14) 1

15) 3

16) True

17) False

18) 3

19) 1

20) 3

21) 3