Create IFC Parameters

This function creates the most common IFC properties often needed in every project.

All IFC software that analyzes data, eg. rulesets in Solibri, depends on standardized IFC Common property sets.

This function creates the shared parameters which are used in IfcCommon property sets. Some of them is exported automatically, such as "LoadBearing" for IfcWall. And some do not export unless you have these parameters.

For example "IfcSite.Name" might be a BIM requirement in your project. Or you might want i just because it looks better. In any case you need a "Project Info" parameter called "SiteName". Simply give the parameter a value and it will export to IFC as such.

- This is the IfcCommon parameters this function creates and how you use them:

  AboveGround,
  EntranceLevel,
  BuildingID,
  BuildingLongName,
  BuildingDescription,
  SiteLandTitleNumber,
  SiteName,
  SiteLongName,
  IfcLongName,
  IfcName,
  IfcDescription,
  IsExternal,
  AcousticRating,
  SecurityRating,
  ObjectTypeOverride,
  ZoneName,
  ZoneObjectType,
  ZoneDescription,
  ZoneClassificationCode,
  ZoneGrossAreaPlanned,
  ZoneNetAreaPlanned,
  ZoneCategory,
  ZonePubliclyAccessible,
  ZoneHandicapAccessible,
  NetPlannedArea,
  IFCExportAs,
  IFCExportType,
  IFC CAD Layer,
  IfcTag

  AboveGround:
  A property to set the IfcBuildingStory.AboveGround. It is found on Level instance Properties.

  EntranceLevel:
  A property to set the IfcBuildingStory.EntranceLevel. It is found on Level instance Properties.
For example checking both parameters in Revit returns both IFC properties "True". Unchecking them will return "False".

BuildingID
A parameter to set Pset_BuildingCommon.BuildingID
In Revit it is found in Manage->Project Information:

IfcName
A parameter to override the IfcEntity.Name property. It could be used for most Revit Categories and also to override IfcEntityType.Name if found on Type Properties. You can expand this to any Category and it will function as an override to Revits family and typenames..

We have chosen to use it on Levels thus setting IfcBuildingStorey.Name.
For instance you may want to display your Level name as "Plan First Floor" on drawings. But still export the Level Name as "01" to IFC if required.
If you want to use it on other categories too just go to Manage -> Project Parameters. Here you can change which Revit Categories this parameter is attached to.

Keep in mind that IFC has both Entity.Name (Instance) and EntityType.Name (Type)
Some viewers like Solibri only displays the "instance" Entity Name. So if you specifically use IfcName on Type Properties then Solibri may not show it.
IfcLongName
A parameter to set the IfcEntity.LongName property.
At this point we have chosen to use it on Levels but it can be used on any Revit Category.
It is found Level instance properties.

SiteName, SiteLongName and SiteLandTitleNumber
These are parameters to set the IfcSite properties.
They are found in Manage->Project Information.

IsExternal
A property to set the IfcXxxCommon.IsExternal property.
It's found on Type Properties of Floors, Doors, Roofs, Columns and Railings.
This is an override, if not set most categories will default to "True" meaning they become External.
For other categories like Walls this is automatically set from the Walls "Function" parameter.

- AcousticRating / SecurityRating
Putting your classifications on these parameters will put them in the standard common IFC property sets.
This is the default location where cost calculation software looks for these properties.

- ZoneName, ZoneName 2, ZoneName 3 etc
The Room parameter "Zonename" allows you to create IfcZone objects.
Each Room can be related to multiple IfcZone categories.
For example "Room 101" can be related to both a security control zone and a acoustic zone.
You could decide you want to use "ZoneName" for security plans and "Zonename 2" for acoustic layouts like this:

This is drawings some need to make anyway.
The benefit of using these parameter names for making them becomes apparent when you export to IFC:
**IfcExportAs**

This type parameter decides the IfcClass on export just as "IFC Export Options" does. It functions as an override to the "IFC Export Options" settings and is found on Type Properties.

Typically you will use the IFC Class -> by Elements which will write to this parameter.

It can be used not just to separate outlets from other Electrical Fixtures like dryers. But to differentiate different types of outlets like this:

- Applicable example values for IfcExportAs:
  - IfcOutletType
  - IfcOutletType:DATAOUTLET
  - IfcOutletType:TELEPHONEOUTLET

If you just rely on the global "IFC Export Settings" then every Category will become the same IfcClass. So using IfcExportAs is a smart thing to do and also allows predefining the IFC Export Class in your familys.

**IfcExportType**

For some system familys like Floors you need to set the enumeration on this parameter. You could use it to set properties like "BASESLAB".

**IFC CAD Layer**

Yes IFC files has layers in the same way as CAD (dwg) files has. This parameter lets you specify the layer for each FamilyType. Tip: You could use the "Combine Parameters" function in Naviate to populate it. It functions as an override to the default settings. If not specified here Revit will use the cad layer table you have selected with the "IFC CAD Layers" function.
- Link to free IFC viewers:
  FZK viewer - simple and fast

  Tekla Bimsight
  http://www.teklabimsight.com/

  Solibri Model Viewer