Inter-Process Communication with Runner

The Settings v2 process uses two way IPC to communicate with the runner process.

Initialization

- On the settings' side, the two way IPC delegates are contained with the `ShellPage.xaml.cs` file. The delegates are static and the views for all the powerToys send the ipc information to the viewmodels as `ShellPage.DefaultSndMSGCallBack`.
- These delegates are initialized within the `Mainwindow.xaml.cs` file in the `Settings.Runner` project.

Types of IPC delegates

- There are three types of delegates for the settings to communicate with the runner:

1. `SendDefaultMessage` - This is used by all the viewmodels to communicate changes in the UI to the runner so that the information can be dispatched to the modules.
2. `RestartAsAdmin`
3. `CheckForUpdates`

Sending information to runner

- The settings process communicates with the runner by using the delegates defined within the `ShellPage.xaml.cs` file.
- Depending on the type of object sending the information, the json is created accordingly.
- If any information has been modified by the user in the GeneralSettings page, then the json file sent to the runner has the name set to `general`, whereas if any information has been modified by the user in any powertoy related settings page, the name of the json file being communicated with the runner is set to `powertoy`.

Receiving information from runner

- The `ShellPage` object has a `IPCResponseHandleList` which is a list of functions which handle IPC responses.

```
// receive IPC Message
Program.IPCMessageReceivedCallback = (string msg) =>
{
    if (ShellPage.ShellHandler.IPCResponseHandleList != null)
    {
        try
        {
            JsonObject json = JsonObject.Parse(msg);
            foreach (Action<JsonObject> handle in
ShellPage.ShellHandler.IPCResponseHandleList)
            {
                handle(json);
            }
        }
        catch (Exception)
        {
```

```
            }
        }
};
```

- Whenever any information is sent from the runner each of the functions in the handle list perform their action on that json object.
- One example of where information sent from the runner is being processed by the settings is in `GeneralPage.xaml.cs` when the user clicks the check for updates button. The information displayed after, such as the user has the latest version installed is a result of this handle.