

## 1 Google Kalender anlegen

Falls nicht vorhanden muss ein Google-Account erstellt und im Kalender die Einträge erfasst werden. Kalendereinträge können auch mittels csv Import eingefügt werden (für weitere Infos siehe <https://support.google.com/calendar/answer/37118?hl=de>)

## 2 Kalender API aktivieren und credentials downloaden

Im Google-Account anmelden und folgenden Link aufrufen:

<https://developers.google.com/calendar/quickstart/dotnet>

Unter „Step 1: Turn on the Google Calendar API“ den Button klicken um die API zu aktivieren und ein File herunterzuladen, das dann die Verbindung zu dem Kalender herstellt. Dieses File ist im .json Format und kann mit einem Texteditor angesehen werden. Punkt C ist zu ignorieren. Stattdessen muss das File `client_secret.json` benannt werden. Der Inhalt darf nicht verändert werden!

Step 1: Turn on the Google Calendar API

Click this button to create a new console project and automatically enable the Google Calendar API:

ENABLE THE GOOGLE CALENDAR API

This opens a new dialog. In the dialog, do the following:

- a. Select + **Create a new project**.
- b. Download the configuration file.
- c. Move the downloaded file to your working directory and ensure it is named `credentials.json`.

## 3 ZIP-File mit Programmdateien downloaden

Die letzte Version mit allen benötigten Files kann von diesem Link downgeloadet werden:

<https://github.com/ramsi05/XmlCreatorForAic/blob/master/zip/XmlCreatorForAic.zip>

Anschließend entpacken und im Unterverzeichnis „XmlCreator“ die Datei `client_secret.json` mit derjenigen aus Pkt. 2 ersetzen.

## 4 Ausführen

Mit der Batch-Datei „StartXmlCreator.bat“ kann das Programm gestartet werden. Hierbei kann in der Datei die Anzahl der angezeigten Termine definiert werden indem nach dem Aufruf

XmlCreatorForAic.exe die Anzahl angegeben wird (z.B. XmlCreatorForAic.exe 10). Weiters ist eine Datei mit dem Namen Input.xml mit der Vorlage für die Anzeige erforderlich (Beispiel in der ZIP enthalten. Siehe auch das XSD der AIC Entwickler:

<http://www.alarm-info-center.at/services/infocenter.xsd>). An diese Vorlage wird im Outputfile eine neue Seite mit den Terminen angehängt.

Nach Ausführung der .bat sollte eine Datei Output.xml und log.txt erzeugt worden sein. Ist die Datei Output.xml nicht vorhanden steht in der Datei log.txt die Fehlermeldung. Danach muss das AIC neu gestartet werden. Hierzu kann die Batchdatei „KillAndStartAic.bat“ verwendet werden. Dazu muss nur mit einem Texteditor der Pfad zum AIC umgeschrieben (Standardmäßig „C:\AIC\AlarmInfoCenter.exe“) und der Benutzer ersetzt (Standardmäßig „Einsatzanzeige“) werden.

## 5 Automatisierung

Um das Ganze zu automatisieren wenn die Anzeige 24/7 läuft kann dies mit Windows Boardmitteln automatisiert werden um z.B. einmal am Tag die Termine zu holen und das AIC neu zu starten.

Dazu wird in der Systemsteuerung unter Verwaltung der Punkt Aufgabenplanung (Windows 7) gewählt und eine neue Aufgabe angelegt. Als Trigger wird der gewünschte Zeitpunkt eingestellt. Unter Aktion wird der Konsole die Batch-Datei übergeben. Dazu die Parameter wie folgt setzen:

Aktion: Programm starten

Programm: Link zur Windows-Konsole (hier am 64-Bit System) „C:\Windows\SysWOW64\cmd.exe“

Argumente: /c “StartXmlCreator.bat“

Starten in: Verzeichnis indem die „StartXmlCreator.bat“ liegt

