

```

local helpers = dofile('documents/lua/json_helpers.lua')local json = helpers.jsonlocal readfile =
helpers.readfilelocal esc = helpers.esc

function generate_personalhort(path)local s = readfile(path)if not sthenreturn endlocal t =
json.decode(s)local p = t.personalorlocal id = 'personal'local kv = string.format("name =
-- debug : write the exact payload to the log before emitting to TeXtexio.write_n('log', 'DEBUG : CV Personal New payload : C

function generate_experienceshort(path)local s = readfile(path)if not sthenreturn endlocal t =
json.decode(s)local ex = t.experiencesortex.print('
section *Experience')local count = 0for ,e in pairs(ex)do count = count + 1if count >
3then break endlocal id = 'exp'..countlocal ach = ''ife.achievements and e.achievements >
0then local parts = for ,a in pairs(e.achievements)do table.insert(parts, esc(a))endach =
table.concat(parts, ', ' )endlocal kv = string.format("company = -- debug : log what we will emit to texio.write_n('log', 'DEBU

function generate_skills(path)local s = readfile(path)if not sthenreturn endlocal t =
json.decode(s)local skills = if t.experience then for ,e in pairs(t.experiences)do if e.skill then for ,s in pairs(e.skills)do skill
true end end end end local skills_list = for ,i in pairs(skills)do table.insert(skills_list, i)end table.sort(skills_list)local skill_line =
table.concat(skills_list, ', ' )tex.print(string.format('
CV Skills Render end

generate_personalhort('assets/data/personal.json')generate_experienceshort('assets/data/experiences.json')generate_skills

```