Timestamp	Your Name	Your role in the industry	In your opinion, what kind of optimization techniques are the most used in the gaming industry?	Which of your previously mentioned techniques do you think contributes to increasing fps the most?	What are the biggest problems you faced while optimizing the games?	What are the biggest optimization mistakes people make?	Extra information on optimization. If you'd like to add something
5/10/2021 15:19:12	Šarūnas Ledas	Co-founder of an indie game development studio	Real-time scenes can be optimized in a variety of ways, some of which include optimization of polygons (vertices) in the scene, number of draw calls (which is usually impacted by the number of separate objects and materials), fill-rate optimization, baking lighting or reflections into textures (lightmaps, reflection probes), optimizing the number of skinned meshes, etc.	Every mentioned element impacts framerate, and it's usually a combination of those that impacts the whole performance. It also depends on a project – whether it's animation heavy, realistic, needs a lot of reclections, transparent surfaces or other qualities. Therefore, every project requires a unique approach and at least general knowledge about every aspect that can impact overall performance.	It's always a challenge to make a scene look unique and/or realistic, because there is always something that has to be simplified or cut altogether.	Probably the biggest mistake is to not take into account all of the mentioned aspects. Because every one of them can be a bottleneck, and usually it's a combination of a few of them.	
5/12/2021 12:20:52	Pascal Diroll	Engineer and Game Design	Texture maps, object combining, pooling, ECS, instancing, occlusion culling	Texture maps, ECS, instancing and occlusion culling	Instanced loading of areas while not active areas still need to process logic	Too big textures, too many objects, unsmart handling of objects active state and lifetime, as well as overclouding update loops	
5/21/2021 15:17:00	Sophie	VFX Artist Intern	polycount reduction, texture atlas, drawcall optimization, LODs, capped particle emissions	Depends on the game. A 1st person shooter benefits less from a LOD system than a citybuilder. But generally spoken, poly reduction is usually the most effective way to safe fps. In effect heavy games, particle caps, low texture sizes and effect LODs are very effective.	Finding balance between optimization and sacrificing the visual outcome. How much can I reduce polys, particle emission, texture size without sacrificing visuals?	especially for artists, its a common mistake to not optimize at all besides reducing polys because they think its none of their business and rather the job of a technical artist/programmers.	