Imagine you are back in 1870 and you are named Thomas Adams. You’re walking through the forest with your employees, the air is hot and humid. You are slapping insects away and gathering the sap from the chicle trees. You are trying all sorts of combinations to invent a « new synthetic rubber », because natural rubber is too expensive. It’s hot and it’s been days that you’re not succeeding.

On top of that, you are a « known inventor » and you work as a secretary for the leader of Mexico, Lopez de Santa Anna. And that sap is just not turning into anything like rubber.

How would you feel ? Everyone expects you to succeed , you have the position, the reputation, the ressources. The more the experiments don’t work, the more you feel it is your fault. You don’t know enough. You didn’t try hard enough.

At that point Adams couldn’t know that months later, by adding sugar and flavours to that sap, he would invent a new chewing gum. Today, millions of people buy the product.

Now fast-forward to 2020. Have we not all felt that way at least once ? How about more than once ?

That is my topic today. I will explore with you what is the imposter syndrome and why is it making the headlines more and more these days.

The symptoms :

1. persistent doubt of your skills
2. fear of being exposed as a "fraud"
3. constant feeling that others « know more »
4. thinking that you do not really deserve achievements

(Even Despite external evidence (success), those experiencing this syndrome remain convinced that they do not deserve all they have achieved and that others certainly know more than them.)

Approximately 70% of all people experience Imposter Syndrome at least once in their life, with entrepreneurs and experts experiencing it at much higher levels.

The causes :

1. Very high expectations and standards from the tech industry and from the media

In coding and, esp data science, we are expected to be constantly aware of what the latest trend is. I guess we have all noticed that this tends to change quite rapidly.

Thus, many people feel « obsolete » or that they are not « up to date » with the latest framework or language.

However, the problem is that :

Focusing on what you don’t know will not let you see what you do know.

Also : Given how often new tech startups get covered in the media and how their founders are portrayed as brilliant and uniquely creative, it’s no wonder that so many people feel that they can never make it as a top-tier developer. Software development also has a mythos that’s grown up around it that says only the super-smart people are able to grasp it.

1. Social Media

Living hooked on social media also during our working hours does not help either. It indirectly triggers us to compare ourselves with other people all the time : be it their positions, their projects or even their code…

1. The mirage of how good others are : this is closely related to point 2. Did you ever see anyone share a professional fail every week on their LinkedIn? Or how many times they copied a code from the web ? That’s rare. Even at work, do you really know how others do ?

And now the greatest paradox of the imposter syndrome : do you really know what others know ?

So how can we fix this ? These are some possible solutions :

1. Accept « feeling uncomfortable ». Coding and data analysis are developing as we speak. We need to accept that this job comes with a need of constantly learning and that no one has « full knowledge » or has achieved the « fixed amount » to be worthy. What was the most performant language last year might not still be in 2020. What the industry uses in data science in Belgium might not be what they’re used to in Morocco.

So we shouldn’t stress but analyze how we can adapt.

1. Start to consider failure as a learning opportunity. We are usually taught that if we do not understand something fast it is because we are not « intelligent enough «If in sports we need to push ourselves to get stronger, and not just sigh that we don’t have « enough muscle », why would not that be true about learning?

Many people are starting to realize that intelligence is not fixed but tied to how we respond to effort and challenges.

1. Figure out how you learn best : is it by videos, mind-maps, or by doing projects, like many coders ?
2. Track your personal progress and compare it with skills, not people. This will help you see your evolution. When you already have a job, don’t be afraid of asking regular feedback about your work and do your best to build constant communication with colleagues and your boss.
3. Use « social media » time for learning a new skill or doing a project with other coders/ analysts. That will create opportunities where you will truly gaining and exchanging knowledge , building relationships rather than comparing appearances.